KEYFRAME

MAGAZINE

THE PUBLICATION FOR LIGHTWAVE, INSPIRE, AND AURA ENTHUSIASTS

The Creature Core Project:

Character Animation from Savage Frog!

L[6] for Mac

EndoMorph Hair

OpenGL Programming

Design and Modeling Basics

Place with computer magazines

07

0 74470 92596 5

Issue #19 July/August 2000
$9.95 US $14.95 Canada

LightWave® [6]
THE JOY OF SIX

Allow yourself to be seduced.

Seduced by the power, the elegance and the flexibility. Your first encounter will reveal why LightWave dominates the demanding world of show business.

LightWave has always been the industry standard for exceptional realism. Today it just got better. Much better. Now, with caustics, radiosity and integrated Hypervoxels, LightWave [6] understands and delivers the nuances of lighting and volumetrics better than any other animation system on the market. The addition of a 96-bit floating-point render pipeline assures you ultimate color depth and accuracy.

LightWave [6] introduces the world's fastest and most accurate hybrid inverse/forward kinematics engine. Also, unique to LightWave [6] are IntelligEntities™. These include Endomorphs™ and Skelegons™ used to create and control character animation with efficiency and fluidity.

With the release of LightWave [6] nothing was left to chance. We completely redesigned the program and the process with you, the artist, in mind. The joy of LightWave, all the features, all the functions — better, faster, fun.

The seduction is complete.

NewTek booth #535 at Siggraph '99

www.newtek.com
Call **(800) 862-7837**
Outside the U.S. call (210) 370-8000

LightWave 3D is a registered trademark of NewTek, Inc. Hypervoxels, IntelligEntities, Endomorphs and Skelegons are trademarks of NewTek, Inc. © NewTek, 1999. All rights reserved.

NewTek

ARETE'S

NATUREFX

FOR LIGHTWAVE 3D

ATMOSPHERE | CLOUDS | WATER | WAKES

L6 Compatibility Guaranteed!

Deep Blue Sea image courtesy Flash Film Works. ©1999 Warner Bros..

AVAILABLE AT:

SAFE HARBOR COMPUTERS

800-544-6599 262-548-8120
www.sharbor.com

CREATED BY

Dynamic Realities

800-944-5520 • 702-990-2277
WWW.DYNAMIC-REALITIES.COM

KEYFRAME
MAGAZINE

Editor
Doug Kelly *dakelly@earthlink.net*

Advertising Rep
Alice Edgin *sales@keyframemag.com*

Co-Conspirators, Writers, and Artists
Paul Davies *pablo@savagefrom.com*
Brenden Mecleary *bman2@ix.netcom.com*
Steve Pugh *steve@foundation-i.com*
Michael Reiser *MikeReiser@aol.com*
David Warner *dwarner@albany.net*

Proofreader
Patty Kelly

The Digital Underground
DMG Publishing
2756 N. Green Valley Pkwy. #261
Henderson, NV 89014
702-990-8656

How To Get It
888-778-WAVE
http://www.keyframemag.com

Keyframe Magazine is published 6 times annually by DMG Publishing. © 2000 DMG Publishing. All rights reserved. Reproduction, in part or in whole, without written permission is strictly prohibited. Printed in the USA.

The Secret Formula
Keyframe Magazine is created using Quark 4.1, Photoshop 5.0, LightWave 6.0, and various plug-ins.

No dongles were harmed in the making of this publication.

Departments

3 Editorial by Doug Kelly

4 Ear to the Wall
- 3D Exchange Workshops
- NewTek at SIGGRAPH
- LightWave6.com
- VRML97 Export Plug-in
- IBM Intellistation
- Intel Itanium

6 On Topic by David Warner
- Radiosity Lighting, Mac Installs, Parts, Subdivision Order

10 Macintosh Up Close & Personal by Jason Goodman
- L[6] for the Mac

26 Reviews
- Worley Laboratories' Taft Collection

32 Final Cut by Steve Pugh
- Hooray for Hollywood!

Tutorials

8 Programmer's Hideaway by Brenden Mecleary
- OpenGL Control Basics

13 Back to the Basics by Michael Reiser
- Modeling Tools & Techniques

16 The Creature Core Project by Paul Davies
- Cover Case Study

19 Character Shop by Doug Kelly
- Growing Hair on an EndoMorph Character

EDITORIAL

Issue #19 July / August 2000

*TALENT NOT INCLUDED

SIGGRAPH was exhausting, but very cool. The tools that are now available to LightWave artists are simply amazing, if you compare them to the state of the art two years ago. Feature film quality effects are now built into our core software, and the third-party plug-in market is showering us with capabilities that even ILM struggled to achieve only a few years ago.

Unfortunately, this cornucopia of wonders can bury you if you're not careful. These tools are seductive, and they can distract you from the core of your work, the necessity of your development as an artist, and the perception of what is really important to you, to your art, and to your career. For what it's worth, here are a few of my thoughts on some crucial issues of being a LightWave artist.

If it were up to me, all software and hardware packaging would carry the warning label, ***talent not included**. Having the latest and greatest tools does not mean you are any better - or worse - than any other artist. It just means that, at one point, you had access to some financial resources.

TANSTAAFL
There ain't no such thing as a free lunch. I get a depressing number of emails from wannabes who think there is some magical shortcut they can take, some hidden switch on the back of the computer or secret panel in Layout, that will instantly make them a great artist. Sorry, Charlie; it's not going to happen, now or anytime. Art grows from highly developed craftsmanship, luck and inspiration. Developing the skills to competently realize your visions takes years of practice. Until you commit to consistent, focused, *daily* practice, you are not even beginning to become an artist.

Caveat Emptor.
I get at least one email a week (usually more) from students who want my advice on what computer to buy or build, just before they start classes at some fly-by-night media school. As if paying the cost of hardware, software and outrageous tuition can buy you out of the requirement for talent and practice. Hot tip: if the school doesn't require a portfolio review, and it teaches only certain software, it shouldn't get your money.

Plan your work; work your plan.
So your goal is to work at Pixar, or ILM, or to have your own studio. Fine; I am not going to try to dissuade you. But you need to have a plan, even a single sentence, on paper where you can read it every day. And every day, you need to look at that piece of paper, then *do something* that gets you closer to your goal.

Talk is cheap.
Don't waste time talking about the great piece of LightWave art you are going to create. There is something about putting visual concepts into words that takes a great deal of the potential out of an essentially visual idea. If you spend too much of your energy talking about art, you won't have anything left with which to create art. We all know blowhards who talk a great game, but who haven't actually produced a piece of worthwhile art in ages (if ever).

Most important of all, **take care of yourself**. It's just art; it's not worth dying, being crippled, or missing out on real life. Try to strike a balance, and remember that **people are more important than things**.

Doug Kelly

Ear to the Wall

News from the LightWave arena

NewTek at SIGGRAPH

NewTek and its products appeared in more than 12 booths at SIGGRAPH 2000.

NewTek's streaming webcast from the booth attracted over 5,000 viewers from around the world, including viewers from Foundation Imaging, WestWood Studios, ILM, LucasFilm, Threshold Digital, and Pixar. NewTek worked closely with NVidia and IBM to create workstations for all NT demonstrations. Apple provided 14 G4 workstations for a StealthNET-controlled render farm at the show. NewTek premiered the first OSX 3D application by demonstrating LightWave[6] Modeler on OSX. LightWave[6] was demonstrated on the new 64-bit Intel chip (code name: Itanium).

Playstation 2 Development

Sony game developers can now get a PS2 converter for LightWave[6]. Currently the exporter includes support for models, lights and textures. Motions and deformations are on the way.

NewTek to Develop Native Version of LightWave [6] and Video Toaster for Intel's Itanium Processor

NewTek plans to develop a native IA-64 version of LightWave[6] and the Video Toaster for the new Intel Itanium processor. The new Itanium processor, the first implementation of Intel's forthcoming IA-64 processor architecture, will offer a high-performance solution for processor-intensive applications such as 3D animation, video editing, and video painting and layering. When Intel's new Itanium processor ships, LightWave [6] and Video Toaster users will have the opportunity to increase their productivity by leveraging the power of these new systems.

"We are excited to be working closely with Intel as both companies move forward in advancing their respective technologies. By forging relationships with companies like Intel we can share information during the development phases to make certain that our respective products compliment each other," said Brad Peebler. "This sort of relationship will provide our customers with much faster, more robust solutions."

"Intel has been working closely with companies like NewTek to ensure that applications like LightWave and the Video Toaster will take advantage of the full benefits of the Itanium processor," said John Thomas, marketing manager of the Digital Media Solutions Group at Intel. "Processor-intensive features like Hypervoxels can take advantage of the Itanium processor's floating point capability, increasing performance, flexibility and scalability for large and complex animations."

3D Exchange Launches New Workshop Series

NewTek's Premier Authorized Training Facility recently announced the official launch of its new Workshop Series. This ongoing series of workshops is designed to provide students and professionals alike with invaluable knowledge in the field of digital content creation. 3D Exchange will bring in some of the industry's top professionals to spearhead the workshops. "We are always looking for new ways to enhance our students' knowledge so that they can go out into the industry and be ready to tackle any task that is asked of them," said 3D Exchange chief operations officer Arthur Howe. "What we are offering here is something that working professionals can benefit from as well, by expanding their knowledge base which in turn makes them more marketable as digital artists." The series of informative workshops will be kicked off with "Building Character" led by Scott Fritts of Threshold Entertainment. This workshop will show attendees industry- proven techniques for 3D character design and realization. This will be followed by workshops covering topics such as character animation, Lscript and Digital Compositing.

PRICING AND AVAILABILITY For a list of scheduled workshops, pricing and seating availability please visit the 3D Exchange web site at **www.3dexchange.com/04_workshop.html** or contact Arthur Howe via e-mail at studentservices@3dexchange.com, via phone at (510)747-1540 or fax at (510)747-1545.

ABOUT 3D EXCHANGE Established September 1996, Alameda, CA-based 3D Exchange is dedicated to providing state-of-the-art training to help each individual develop the skills needed to advance their career in the 3D animation industry. You can contact 3D Exchange at 2100 Ferry Point, Suite 100, Alameda, CA 94501 and by phone (510)747-1540 or fax (510)747-1545 or by visiting the website at **www.3dexchange.com**.

Graphic Board Certification

NewTek has recently started a Graphic Board Certification program. NewTek is working closely with leading graphics cards vendors (including 3dfx, 3DLabs, ATI, ELSA, Evans and Sutherland, Intense 3D, Matrox and Nvidia) to make sure that everything from lens flares to the various display and texture modes look great, and to help design specialized drivers that take full advantage of LightWave [6]. Upon acceptance to the program, vendors will receive a LightWave certification with a special LightWave [6] certified logo for use in their packaging as well as collateral materials.

LightWave6.com Updated

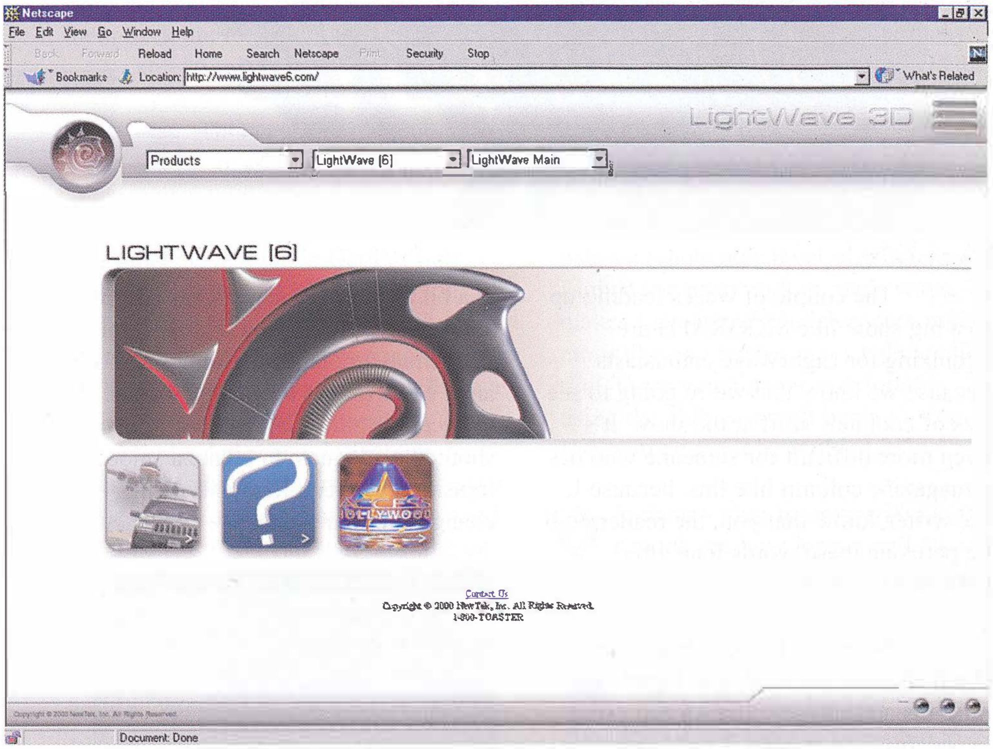
NewTek's website for LightWave [6] has a facelift and a lot of new content. Check it out!

LightWave [6] and the Web

Did someone say Internet? There are several new ways to get LightWave [6] data onto the web in real-time. If you are a web kind of person, check out Hypercosm, VRML, and Imagination in Motion for some of the latest solutions.

VRML97 EXPORTER

A new beta version of the VRML97 Export plug-in is available for the Intel and Macintosh platforms in the Downloads -> Plug-ins area of the NewTek website:
www.newtek.com/lightwave/downloads/plugins/plugins.html


Netscape
File Edit View Go Window Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://www.lightwave6.com/
What's Related
LightWave 3D
Products
LightWave [6]
LightWave Main
LIGHTWAVE [6]
HOLLYWOOD
Contact Us
Copyright © 2000 NewTek, Inc. All Rights Reserved.
1-800-TOASTER
Document: Done

NewTek Announces LightWave 6.1

At SIGGRAPH, NewTek announced LightWave 6.1, a significant upgrade to its popular 3D animation software. LightWave 6.1 integrates core components such as a particle system and the Motion Designer soft-bodies dynamics engine, while also improving upon the software render speeds, mesh deformation performance and overall usability, while also adding hundreds of new features.

"LightWave 6.1 is a true testament to our new architecture, as well as to the serious commitment we have to our user base," said Brad Peebler. "The new architecture under 6.0 has allowed us to integrate features faster than ever. What would normally be a small service update is actually a giant feature upgrade as well. That is what the new system was designed to do, let us add features fast! As far as customer commitment, we are giving this update away to all registered 6.0 users. At NewTek we think of customers as partners. This year we are really going after customer relations in a big way. 6.1 is a great example of that attitude."

"We had to fly in extra ass, because LightWave 6.1 kicked all of it," said Steve Pugh of Foundation Imaging.

Continued on page 18

IBM and NewTek plan strategic marketing for 2000 and 2001

NewTek recently announced that the company and IBM plan to work together for the remainder of 2000 and into 2001 to promote broadband Internet digital content delivery as well as integration of 3D and video tools. At SIGGRAPH, NewTek and IBM worked together to showcase the power of these technologies by demonstrating key NewTek products on IntelliStations as well as by producing live streaming "TV" segments from the show floor. This content was online for the world to view through Internet broadcasting. The programs were produced on the exhibit floor and web cast using NewTek tools including LightWave [6] and Video Toaster [2].

"The implications for the film, video, Internet, and broadcast industries are very exciting," said Jay Ornellas, IBM global manager, digital content creation. "Advanced capabilities will be available to users at an astonishing price point on the industry standard platform." IBM is an authorized provider for all of NewTek's products.

"The NewTek suite of products dovetails perfectly with IBM's suite of products for the film, video and broadcast industry," said Brad Peebler, NewTek's vice president of marketing of 3D graphics. "The IntelliStation family makes Video Toaster [2] and LightWave [6] look and run really well, and these systems are aimed at the same people who are customers of NewTek."

For more information, visit the IBM website at http://www.ibm.com or go to the IBM IntelliStation website at http://www.pc.ibm.com/intellistation.

Radiosity Lighting, Mac Installs, Parts, Subdivision Order

by David Warner

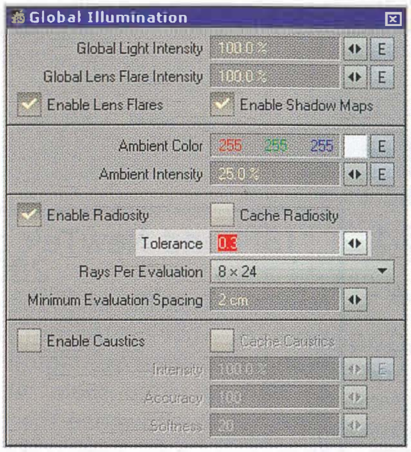
The couple of weeks leading up to a big show like SIGGRAPH are agonizing for LightWave enthusiasts because we know that we're going to see lots of cool new stuff at the show. It's even more difficult for someone who has a magazine column like this, because I, the writer, know that you, the reader, will be perusing these words long after SIGGRAPH 2000 is history.

Normally I would banter on about some of the cool new LightWave plug-ins and add-ons that we can expect to see unveiled at SIGGRAPH this year, but by the time you read these comments, you will already know much more about these things than I know as I write this column. So this time I'm going to give my fingers and your eyeballs a rest, and just get to the meat of this column.

With the omission of my normal ramblings and with a bit of extra space that the folks at Keyframe have given me for this issue, I can squeeze in a whole bunch of L6 questions and answers from the LightWave mailing list.

Q I've set up an animation using Radiosity and each rendered frame looks really great, but when the animation is put together and played back, the Radiosity lighting seems to flash and jerk around in a way that is unacceptable. Is there a way to fix this?

A What is happening in your animation is that the Radiosity Tolerance levels are set too low, so the reflected light beams are too sensitive to changing conditions in your scene. The default Radiosity Tolerance value of 0.3 is a bit too low for many scenes, so the solution is to simply increase the value a very small amount. The Radiosity Tolerance value ranges between 0.0 and 1.0, so increasing this value to 0.5 or above should be adequate to create a smoother transition between Radiosity light changes in your scene.



Global Light Intensity 100.0 %
Global Lens Flare Intensity 100.0 %
☑ Enable Lens Flares
☑ Enable Shadow Maps
Ambient Color 255 255 255
Ambient Intensity 25.0 %
☑ Enable Radiosity
☐ Cache Radiosity
Tolerance 0.3
Rays Per Evaluation 8 × 24
Minimum Evaluation Spacing 2 cm
☐ Enable Caustics
☐ Cache Caustics
Intensity 100.0 %
Accuracy 100
Softness 20

***Figure 1** The Global Illumination panel with the Radiosity Tolerance field highlighted.*

Q I know that LightWave 5.6 and 6.0 will peacefully co-exist on a Windows machine, but what about the Mac versions of these programs? Do I need to do anything special to have LightWave 5.6 and L6 installed on my Mac?

A As long as you do not have one of the new hardware keys that connects via the USB port, you will be able to have LightWave 5.6 and L6 installed on the same machine without any problems. The trick is to make sure that you install L6 in a directory that is different from where you already have LightWave 5.6 installed. If you allow the L6 installer to select an install folder for you, it will most likely overwrite the LightWave 5.6 program data and this will, of course, prevent version 5.6 from working properly.

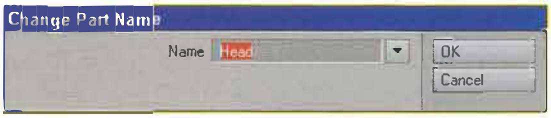
Once you have LightWave 5.6 and L6 installed in separate folders, make sure that you have appropriately named aliases set up to point to each version of Layout and Modeler. One thing you may notice is that the file type/creator info in your Mac system will be updated so that the L6 applications are used for all of your scene and object files. However, you will still be able to directly load scene and object files into LightWave 5.6, unless the scene and object files have been created or modified by L6. L6 scene and object files are not backward compatible with older versions of LightWave.

Q I have a texture that I need to move. In LightWave 5.6, all I had to do was go into the Texture panel, apply a Texture Velocity in an appropriate direction, and the texture would move. I no longer see the Velocity options available in the Surface Editor panel, so how do I move a texture with L6?

A The Texture Velocity function has been removed because you can now apply an envelope to the texture's Position, so the Velocity options were no longer necessary. With a Position envelope, it is very easy to keyframe the location of a texture over time. This can be especially handy if you are trying to make a moving seamless texture "loop" over time. Of course, an even easier way to move a texture is to simply assign a Reference object to that texture, then keyframe the texture's position by moving the Reference object.

Q What is the benefit of having the ability to define Parts of an object, as opposed to the 'old' way of defining Surface names for different parts of an object?

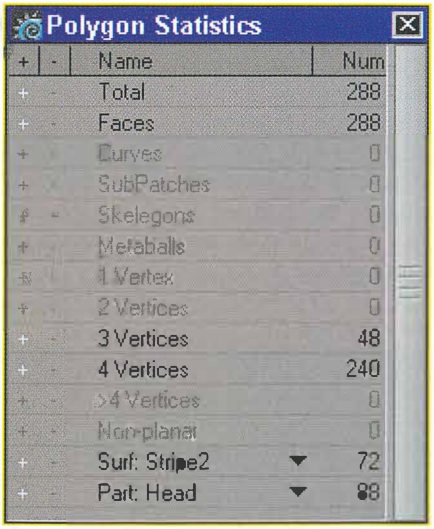
A Before L6 came along, you could define Surface names for groups of polygons, but if you wanted to work with a group of different surfaces, you needed to select them individually and hope that you didn't forget any. With L6, you can now define entire groups of polygons or surfaces as a Part and then quickly select that group of polygons to work with later.


Change Part Name

Name: Head

Figure 2: The Change Part Name tool allows you to define parts of an object that are separate from polygon Surfaces.

For instance, if you were to create a bunch of interchangeable body parts for some character animation work, you could define a Part named "Head" and within that part could be dozens of different polygon surfaces for the character's eyes, hair, lips, teeth, tongue, skin, etc. So, instead of having to select all of the polygon surfaces that make up the head of your character, you could instead bring up the Polygon Statistics panel, select the Part named "Head" and highlight all of those polygons very easily.


Polygon Statistics

+	-	Name	Num
+	-	Total	288
+	-	Faces	288
+	-	Curves	0
+	-	SubPatches	0
+	-	Skelegons	0
+	-	Metaballs	0
+	-	1 Vertex	0
+	-	2 Vertices	0
+	-	3 Vertices	48
+	-	4 Vertices	240
+	-	>4 Vertices	0
+	-	Non-planar	0
+	-	Surf: Stripe2	72
+	-	Part: Head	88

Figure 3: The Polygon Statistics panel with Surfaces and Parts shown near the bottom.

Note that, unlike Polygon Surface names, Part names are not required for the different parts of your objects. Part is merely an organizational tool that will help you work more efficiently if you do a lot of recycling of object parts, or if you just want to save yourself some time while modifying large groups of polygon surfaces.

Q I've noticed that the Get and Put functions no longer exist in L6 Modeler. Other than saving and loading the individual object files, how can I get objects into Modeler and put them back out to Layout the way I was able to with LightWave 5.6?

A This is where the Hub, one of the coolest but least understood new L6 features, comes into play. The Hub serves as sort of a full-time automatic Get/Put link between Layout and Modeler. If you have a scene already loaded into Layout when you enter Modeler, then to the left of the different layer tabs, you will see a pull-down list of all the objects in the currently loaded scene. In fact, upon entering Modeler, the first object in your scene should automatically be displayed in Modeler so you can start working on the object immediately, without having to Get anything from Layout. However, if you want to work with another object in the Layout scene file, all you have to do is click on the pull-down list of available objects and select the one you want to work with. This process is completely automated, so you don't have to worry about clearing out the object currently loaded in Modeler; you just select the object you want to work with and it appears.

Of course, as you make changes to the various objects in your scene, you will need a way to Put those changes back out to Layout. One way is to use a function very similar to the Put command. If you look in the upper-right-hand corner of the Modeler interface, you will see a small arrow pointing downward and, just like any other pull-down menu, when you click on this button a list of options will appear. The "Send To Layout" option is exactly like the old Put command, where the current Modeler object will be sent to Layout. The "Switch To Layout" option will send you back to the Layout interface to continue working on your scene. The most interesting of all these options is "Synchronize To Layout" because it will actually establish an automatic link between Modeler and Layout, so that any changes you make to the object in Modeler will automatically be updated in Layout without you having to go through any extra steps.

When the Get and Put functions first appeared in LightWave 5.5 many years ago, everyone was quite excited about how much time it would save, and how this simple idea would remove some tedious steps you had to go through every time you wanted to edit an object in a scene. Well, the Hub establishes a whole new level of efficiency for LightWave users and, in many ways, makes Layout and Modeler seem much more like two halves of a single application. Once you become comfortable using the Hub, you will begin to wonder how you ever got by without it!

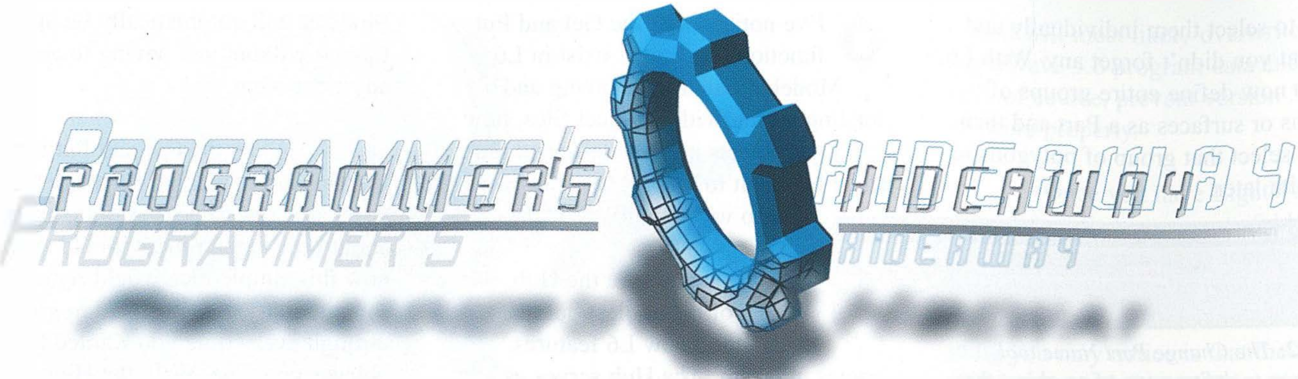
Q Is there a plug-in or program that can tell me how long it will take to render an animation before I actually start rendering it?

A No. A few applications have been released on the Internet for free download, but it just isn't possible to make even a remotely accurate estimate of how long it will take LightWave or any 3-D animation program to render a scene. This is because of the very nature of 3-D rendering. The computer must do millions and millions of calculations to determine the path a ray of light takes and how that ray of light will affect objects in the scene. If you have only one light in a scene and you do not have any ray-tracing options turned on, then it may be possible for a program to come up with some sort of time estimate that is close to the final render time, but very rarely will anyone set up a scene with only one light.

The only real way to even come close to a real-world estimate of an animation's render time is to do a very low-res preview render of the complete animation. If you cut the resolution down to 1/4 or 1/8 of the final render size and turn down (or turn off) the anti-aliasing options, then you should be able to get a good idea of the final render time for the entire full-resolution render. It is actually a VERY good idea to do this before launching a final render of your work,

Continued on page 29

OpenGL Control Basics



by Brenden Mecleary

Although the OpenGL control has been present in LWPanels for quite awhile, I haven't gotten around to using it until recently. Chalk it up to laziness; I didn't *need* it and I wasn't feeling like learning a whole new API without a good reason. So when I had to learn OpenGL in the course of my work in the last month or so, I decided to have another look at the panels control. I found that the GL control is great for drawing 2D or 3D quickly to the screen. In this column I'll explain how to draw a logo bitmap, but you can certainly go much farther.

If you aren't already familiar with OpenGL, it is a platform-independent API for drawing graphics. The "platform independent" line isn't BS, either. It actually works that way. OpenGL is relatively easy to grasp, and does quite a few nice things like clipping and projection for you. However, it doesn't do *everything* for you. OpenGL basically just renders polygons. Consult a good reference book such as *The OpenGL Programming Guide* (Addison-Wesley, ISBN 0-201-63274-8) for a more detailed explanation of OpenGL.

Using the OpenGL Control

To begin using the OpenGL control, include the appropriate GL headers. You don't need these to *create* a GL control, but you need them to do anything useful with it. So on Win32, include `<Windows.h>` - yes, yes, the root of all evil, I don't wanna hear it - and `<GL\gl.h>`. To use GLU, the OpenGL Utility library, also include `<GL\glu.h>`. Link with Opengl32.lib. Create your OpenGL control with the `OPENGL_CTL` macro in `<LWPanel.h>`. Install a user *event* function - *not a user draw function* - and you're ready to begin.

Your event handler will be where you do all your drawing. For some reason, draw handlers don't get called properly for OpenGL controls. So start by clearing the background. To do this, call `glClearColor(0.0, 0.2, 0.4, 0.0)`, and then `glClear(GL_COLOR_BUFFER_BIT)`. OpenGL is a state-based API. You tell it something once - say, the current color - and it continues using that until you tell it to do something else. OpenGL has several other buffers available, but right now you just want to clear the color buffer. In practice, you usually will not need to clear the background. If you draw a polygon that completely covers your control, clearing the background is a waste of time because the background will never be seen anyway.

Now that your background is erased, set up your projection matrix. This is the matrix that OpenGL uses to transform points in 3D to points on the screen. The following function calls will set up your projection matrix:

```
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glOrtho(0.0f, 1.0f, 1.0f, 0.0f, -1.0f, 1.0f);
```

OpenGL has several different matrices upon which you can operate. The first call to `glMatrixMode()` tells OpenGL that you intend to modify the projection matrix. Calling `glLoadIdentity()` "resets" the matrix back to a default (though not necessarily useful) value. This becomes important when you repeatedly call functions that operate on matrices. For example, if you zoom the camera in by ten units every frame, eventually you'll zoom the camera way too far in. (That was just an example - don't go looking for `glZoomCamera()`; there is no such function.) To prevent these problems, call `glLoadIdentity()` to clear a matrix before you muck with it. Finally, the call to `glOrtho()` instructs OpenGL to set up an orthographic projection matrix (one that has no perspective).

Take note of this potentially confusing trait of OpenGL projection. By default, the origin (0,0) is at the bottom-left of the control, *not* the top-left like all us cool people use. This makes some degree of sense; points that are higher on the Y-axis display higher on the screen (Think of it like a piece of graph paper). However, for our purpose - 2D drawing - the default projection is just a pain. The projection matrix I described above will move (0,0) to the upper-left of your control, and (1,1) to the lower-right of your control. You can experiment with the parameters to `glOrtho()` on your own.

Now that you can draw, draw a polygon:

```
glBegin(GL_QUADS);
        glColor4f(1.0f, 0.5f, 0.5f, 0.5f);
        glVertex3f(0.05f, 0.05f, 0.0f);

        glColor4f(0.5f, 1.0f, 0.5f, 1.0f);
        glVertex3f(0.05f, 0.95f, 0.0f);

        glColor4f(0.5f, 0.5f, 1.0f, 0.5f);
        glVertex3f(0.95f, 0.95f, 0.0f);

        glColor4f(1.0f, 1.0f, 0.5f, 1.0f);
        glVertex3f(0.95f, 0.05f, 0.0f);
glEnd();
```

The above code begins drawing rectangles, then spits out four points that define a rectangle, indented just a little bit from the edges of the screen. You cannot call all OpenGL functions between calls to `glBegin()` and `glEnd()`. Between those functions, you're expected to draw all your polygons and not do much else. Certain OpenGL functions will fail if called between `glBegin()` and `glEnd()`.

`glColor4f()` changes the current color before each point is created. If you are drawing a flat-shaded polygon, you can call it just once. There are several versions of `glColor*()`. This one takes four floats as arguments ("4f"). They are the red, green, blue, and alpha values for each point. Later on, you'll enable alpha blending so you can see the effects of the alpha value.

`glVertex3f()` creates a new point. You begin a certain type of geometry - say, `GL_TRIANGLES` or `GL_POLYGON` - then feed OpenGL with points until you've created all the geometry you need. The three parameters to `glVertex3f()` are, not surprisingly, the X, Y, and Z positions of the vertex.

To get GL to spit out your polygon, call `glFlush()`. This will force it to complete any pending drawing.

Adding Textures

Right now, you should have a pretty gradient-color polygon on top of a pretty solid-color background. I know you're thinking that this is already even *better* than sliced bread, but wait, it gets even more thrilling. We're about to add a texture map to that polygon.

Start by calling `glEnable(GL_TEXTURE_2D)`. This enables textures. (Duh.) OpenGL allows you to 'bind' textures to numeric IDs. Each texture 'name' is a non-zero integer. The following code initializes one texture for your polygon and should only be run once:

```
        //glImageName is a global unsigned int
        //glImageBytes contains the texture data
        glGenTextures(1, &glImageName);
        glBindTexture(GL_TEXTURE_2D, glImageName);
        glTexParameteri(GL_TEXTURE_2D,
GL_TEXTURE_MIN_FILTER, GL_LINEAR);
        glTexParameteri(GL_TEXTURE_2D,
GL_TEXTURE_MAG_FILTER, GL_LINEAR);
        glPixelStorei(GL_UNPACK_ALIGNMENT, 1);
        glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, 32, 32,
0, GL_RGB,
                GL_UNSIGNED_BYTE, glImageBytes);
```

First, the call to `glGenTextures()` tells OpenGL to allocate one texture name and store its value in glImageName. `glBindTexture()` makes that texture the current texture. The calls to `glTexParameteri()` tell OpenGL what to do when it needs to magnify or shrink the image. Common choices are `GL_NEAREST` and `GL_LINEAR`. Basically, `GL_LINEAR` works like the "Pixel Blending" checkbox in LightWave. Choosing `GL_NEAREST` is roughly equivalent to turning off the "Pixel Blending" checkbox.

The call to `glPixelStorei()` tells OpenGL the format of the image data you are going to give it. In this case, you tell OpenGL that your pixels are *not* aligned on any particular boundary. Some image formats pack rows to a certain size. For example, in Windows BMP files, each row of pixels must be padded to a multiple of 2 or 4 bytes. (I can't remember which.) Calling `glPixelStorei()` allows OpenGL to correctly read your pixels.

Finally, calling `glTexImage2D()` makes your image bytes into an actual image that OpenGL can use. (See the documentation for an explanation of all those parameters.) Be sure to call `glDeleteTextures()` before your plug-in exits, to free the texture you allocated.

Now you have an image that OpenGL can display and stuff onto a polygon. So add a call to `glBindTexture()` before `glBegin()`. Now, before every vertex, call `glTexCoord2f()` to set the UV coordinates for that point. Note that UV coordinates range from (0,0) at the *upper-left* to (1,1) at the *lower right*.

```
        glBegin(GL_QUADS);
                glTexCoord2f(0.0f, 0.0f);
                glColor4f(1.0f, 0.5f, 0.5f, 0.5f);
                glVertex3f(0.05f, 0.05f, 0.0f);

                glTexCoord2f(0.0f, 1.0f);
                glColor4f(0.5f, 1.0f, 0.5f, 1.0f);
                glVertex3f(0.05f, 0.95f, 0.0f);

                glTexCoord2f(1.0f, 1.0f);
                glColor4f(0.5f, 0.5f, 1.0f, 0.5f);
                glVertex3f(0.95f, 0.95f, 0.0f);

                glTexCoord2f(1.0f, 0.0f);
                glColor4f(1.0f, 1.0f, 0.5f, 1.0f);
                glVertex3f(0.95f, 0.05f, 0.0f);
        glEnd();
```

You now have a textured polygon. Note how much faster it draws than if you had rendered the texture onto an `AREA_CTL`. Note the nice smoothing on the texture. Note that if you move the vertices of your rectangle around, the texture follows and distorts correctly. And, I'm spent.

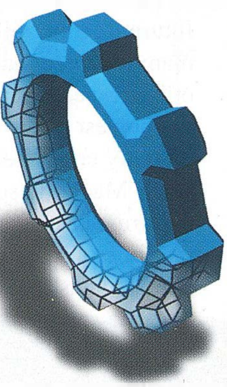
Miscellanea

Enabling alpha blending in OpenGL is even easier than adding texture maps. Call `glEnable(GL_BLEND)`. Then call `glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA)`. The final parameter to `glColor4f()` is the alpha value that OpenGL will use. 0.0 is transparent, and 1.0 is opaque. The OpenGL blend function is good stuff. You can create many effects with it. For example, an additive effect is simply a matter of calling `glBlendFunc(GL_ONE, GL_ONE)`.

When working with OpenGL, keep in mind that its coordinate system is reversed from that of LightWave. Positive Z points *out of* the screen, *not* into it. Positive X is to the right, and positive Y is up.

If nothing else, the OpenGL control can be good for drawing big logos fast. You could also draw previews pixel-by-pixel into memory, then render them as a texture on a polygon much more quickly than you could draw them directly using the drawFuncs. And OpenGL would save you a massive amount of work if you really wanted to do a sophisticated 3D preview. Give it a try.

Brenden Mecleary finally quit Asheron's Crack this week. The other games were getting jealous. They told him so. Signup for the survivors group is at bman2@ix.netcom.com.



Up Close and Personal

by Jason Goodman

Mac LightWave enthusiasts have reason to rejoice. The state of the art in LightWave 3D animation has arrived on the Mac, and it has done so in full force. L[6] is a huge improvement over all previous versions of LightWave in many respects. NewTek has been very vocal about the new program's extensibility and improved architecture. The interface is totally upgraded in both look and functionality. If you have been reading about L[6] on the PC, you know that the laundry list of new features goes on and on. The new capabilities of 6 make it the de facto champ for 3D modeling, animation and design on the Mac. Mac L[6] continues to astonish with its performance as it proves the Mac is a formidable 3D contender. This is only punctuated by such news items as Alias|Wavefront's announcement of Maya for Mac OS X. While this announcement may not have as much to do with LightWave as it does with improvements to the Mac and its future outlook, I think LightWave does deserve some of the credit for proving the Mac's effectiveness as a 3D platform for the last four years.

Current limitations to Mac OS 9 present us with some unique problems when it comes to running LightWave. The Mac does not handle RAM allocation in the same way as do current Windows operating systems. To run L[6] in all its glory, (Modeler, Layout and the Hub running simultaneously) you will realistically need between 384MB and 512MB or more of RAM to be comfortable. Nowadays that may not be unrealistic, but in many cases it will be the impetus for users to upgrade their machines.

Speaking of upgrading machines, this brings us back to a topic I explored one year ago. That is the current state of Mac OS upgradeability. When Apple came out with the first round of PPC 604 based Macs, they introduced a new motherboard architecture. By putting the CPU on a removable card, they promised future upgradeability via low cost CPU replacements. While many of these machines are beginning to show their age in other respects, for the most part users are able to keep pace with the newest levels of performance. As I did one year ago, I recently conducted some render tests with various upgraded and stock Macs and some of LightWave[6]'s new rendering features. Last year's CPU upgrade shootout pitted the fastest (at the time) G3 processors from the top Mac third party developers against one another, and against multiprocessor PCs and DEC Alphas. The results were surprising to some, but in many cases the Mac held its own and even proved to be the fastest platform. With PCs pushing the GHz envelope, and DEC Alphas not currently a viable L[6] platform, the question is open. Will the Mac retain its reputation as LightWave Rendering King or will it slip to second banana?

NewTek seems to be wisely focusing on their core platforms. It was impressive that LightWave 5+ was able to exist in so many environments (Intel, Alpha, Sun, SGI and Mac), but support for the peripheral platforms suffered. NewTek has unofficially stated that a DEC Alpha version of L[6] is in the works, but no release date has been announced. Not much has been said about Sun, SGI or any other OS. If NewTek would forget about all those other platforms and concentrate on Windows and Mac, it could utilize its resources more effectively and, I hope, provide us with a more stable, better supported product.

As an owner of multiple DEC Alpha machines I would be somewhat disappointed to see Alpha L[6] go unrealized, but I would not be surprised. Microsoft has no plans for Alpha Win2k, and most third party manufacturers have not revved their Alpha drivers in some time. As always, economics dictate the viability of a particular platform. I would think that most professional houses would rather pay the alarmingly low prices for faster, newer hardware and cross-grade to L[6] for Intel or Mac then wait around for Alpha software and drivers that will most likely never arrive. I understand that Foundation Imaging has recently upgraded a significant number of machines to graphite G4 Macs. That in itself is a strong win for Mac L[6].

I will not go into too many details about LightWave 6, as most people have read about or seen the PC version by now. I will, however, address Mac-specific issues and problems and render times. Let's take a look at some of the test results.

The machines in question were the same culprits from last year's round up, plus or minus a few players. Since the last test, one of our PowerComputing machines has met its demise. Still alive and well are a Umax s900, a PowerMac 7500/100, a PowerBook G3/400, a 350mhz iMac and a 233hmz iMac. The Umax machine was upgraded with a NewerTechnology 450mhz MaxPwr G4 and the 7500 was tested with both an XLR8 350 G4 and a Sonnet Crescendo G4 450. According to NewTek programmer Matt Craig, some aspects of Mac L[6] have been optimized for the vector processing capabilities of the G4's so-called velocity engine. Due to the common code base of LightWave for the PC, many things could not be optimized. As a result, there is some unrealized potential in the L[6] + G4 equation, but I suppose we will never see it. I pitted these Macs against some multi-CPU Pentium III-based PCs, and the results were interesting, if nothing else.

First, let's examine the complexity (or lack there of) in upgrading your Mac. The hands-down winner in this arena is NewerTechnology. Their G4 upgrade cards require no software, no dip switches and very little in the way of headaches. The tested cards from XLR8 and Sonnet also installed easily and without event, but they did both require that a software extension be installed before the card would function correctly. They also had dip switches on them, but none had to be adjusted in either of my 50MHz motherboard Macs. During the process of this review, NewerTechnology released a utility that installs some monitoring software and gives the user easy access to the Altivec extensions contained on the Mac OS 9 CD (normally installed only on a G4 and therefore not present on your 'to-be-upgraded' machine).

Well-known LightWave animator Ken Stranahan happened to be in New York recently and was visiting MSD, working with us on a LightWave project. Ken was floored by the fact that during normal production (actually during an all-out cross-platform render with 5.6) we were dropping machines out of the queue, upgrading their CPUs, and putting them right back to work with no fuss and only a few moments lost. Try that with a PC; it's a good bet you would be tied up for a while.

The cards were all easy to install and once inside, totally transparent to the user. However, the proof is in the render time. Speeds of the upgraded Macs were impressive. Below is a chart comparing the render times of the upgraded G4 Macs with some stock G3 machines and PCs of various speeds and configurations. Unfortunately, I did not have access to any Apple-branded G4 machines, which would have the significant added advantage of a 100MHz system bus. The upgraded Macs all had 50MHz bus speeds.

The test scene used here consisted of a VW bug model with a polygon count of 18,689. We ray trace rendered the scene once without radiosity and once with. Radiosity, as you may already be aware, is a CPU-intensive type of rendering that creates much more realistic lighting effects. In the case of this particular scene it does not affect the look of the final image too much, but the calculations are still taking place.

It is interesting to note that the multiprocessor PC did have a slight edge over the Mac in radiosity rendering, but it also had two 700MHz processors kicking versus the Mac's single 450MHz CPU. It is always difficult to make these comparisons, as the latest and greatest hardware is constantly changing. Comparing a single 450MHz PIII to G4, the Mac tidily beats the PC. Keep in mind the 450MHz Mac in this case has a system bus that is 50% the speed of the 450MHz PC. Apple is expected to announce faster G4s and multiprocessor boxes soon; that should bury Pentium III-based machines.

NewTek has given Mac users a great leap forward in 3D capability. However, there are some aggravating problems with Mac L[6]. Many of them relate to cross-platform issues, others are just plain annoying. Cross-platform Screamernet rendering is not currently possible in L[6]. In issue #9, we unlocked the cryptic mystery behind the harmonious marriage of Mac, PC and DEC Alpha in a mixed OS LightWave render farm. While most people probably do not have as eclectic a mix of hardware as we do at MSD, it was still a comfort to know that LightWave 5.6 could carry out this feat. In LightWave [6], this capability is inhibited by what seems to be a slight bug related to NewTek's new (and much improved) file hierarchy interpretation. By the time you read this, NewTek may have released a functional version of the appropriately named StealthNet. As of this writing, it is unknown whether this new addition to LightWave will solve the cross-platform render issues.

StealthNet has been flying too low for radar since the L[6] Beta Force began testing of the PC version. The beta list

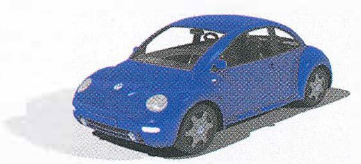


Figure 1: Radiosity on.

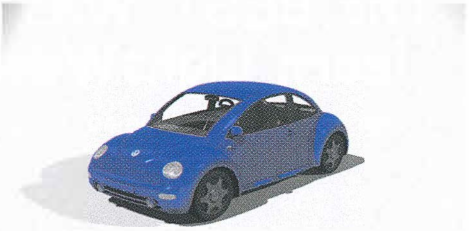


Figure 2: Radiosity off.

Custom built Dual 700mhz Pentium III
100mhz system bus with 2 threads
Test scene = **1m 21s**
Test scene with radiosity = **51m 12s**

Custom built Dual 700mhz Pentium III
100mhz system bus with 1 thread
Test scene = **2m 24s**
Test scene with radiosity = **1h 3m 21s**

Custom built single 450mhz Pentium III
100mhz system bus with 1 thread
Test scene = **3m 54s**
Test scene with radiosity = **1h 26m 10s**

Umax s900 with
NewerTechnology 450mhz G4
Test scene = **1m 55s**
Test scene with radiosity = **1h 9m 14s**

PowerMac 7500 with
Sonnet Crescendo 450mhz G4
Test scene = **1m 53s**
Test scene with radiosity = **1h 12m 25s**

PowerMac 7500 with
XLR8 350mhz G4
Test scene = **2m 28s**
Test scene with radiosity = **1h 22m 33s**

PowerBook (Wallstreet) 400mhz G3 with
100mhz system bus
Test scene = **2m 13s**
Test scene with radiosity = **1h 20m 0s**

iMac 350mhz G3 with 100mhz system bus
Test scene = **2m 35s**
Test scene with radiosity = **2h 21m 5s**

iMac 233 G3 with 66mhz system bus
Test scene = **3m 46s**
Test scene with radiosity = **2h 59m 34s**

PowerComputing PowerBase200 250mhz G3
Test scene = **4m 23s**
Test scene with radiosity = **4h 47m 59s**

seemed to indicate that most anyone who tried could not get StealthNet to do anything. However, its promised capabilities do sound exciting. According to NewTek, StealthNet will utilize TCP/IP to allow administration of render farms over the Internet, among other things. This does make me think that utilizing many, many different machines in different locations will become more common. This would also seem to indicate an increased likelihood of cross-platform situations. I hope that StealthNet will eliminate the cross-platform problems we encountered in L[6].

Another problem that seems like a holdout from 5.6 has to do with opening scene files created on the Mac. The PC cannot do it directly. THIS IS INFURIATING! I actually had a brief email correspondence with Matt Craig about just this issue. According to Matt "PCs cannot handle line endings from different platforms." This is odd to me. I am not a programmer, but it seems that Microsoft Word documents created on the Mac can open on the PC without a problem. You can get a Mac scene file to open on a PC if you open the scene in a text editor and resave it. But that is ridiculous! This is a problem that has existed since 1996!!!

Another slight annoyance pertains to file extensions. In apps like Photoshop, the user has the choice as to whether file extensions are saved with the file. LightWave offers no such choice. You must manually append the file extension (.lwo, .lws, etc.). This makes keeping track of Mac-created files on a PC a problem.

As far as interface customization goes, the Mac and PC cannot load each other's Menu config files. This would be a problem for cross-platform users who have heavily customized their UI and want to have it standardized across an entire studio. Interestingly enough, the Mac cannot load a Menu config file with the .cfg extension at all, even if that file was made and saved on another Mac.

Small issues like this make LightWave look a lot less polished than it should. If you do open a LightWave [6] scene file in your favorite text editor, you will notice that the paths for objects, images, etc. are given in a relative format, such as objects/ball.lwo. This appears to be a PC naming convention, and remains the same even if the scene is saved on a Mac. However, if an image (or object, etc.) is not located in the content directory or in a subdirectory thereof, the scene file

Continued on Page 30

TUTOR VISION 3-D CD

The BEST WAY to learn LightWave!

Easier to understand than any book...
...more tutorials than any video-tape!

FREE Tutorials!

Numerous tutorials written by experts from around the world! Tutorials are in standard HTML and streaming video formats! They're ONLY available from:
www.TV3D.com

FREE Sampler CD!

This FREE SAMPLER CD introduces you to the TutorVision3-D series of CD-based video tutorials. This CD contains 1 complete video and text tutorial from each of the four currently available volumes of TutorVision! It also contains 2 brand new video tutorials not found on any other TutorVision3-D CD! That's 10 video and text tutorials for **FREE!**

5 GREAT CDs to choose from:

TutorVision3-D CD Volume 1
The FIRST interactive video tutorial CD ever created for LightWave! This CD contains nearly 4 HOURS of video tutorials that focus on doing sci-fi and more practical effects. Created with LightWave v5.0, these techniques still work great! PLUS: An interview with Alan Chan, dozens of LightWave renderings, and many objects & texture maps!
Available NOW at a REDUCED PRICE of ONLY $29.95!

TutorVision3-D CD Volume 2
The most popular and most comprehensive tutorial CD ever created to teach Character Animation for LightWave! This 2-DISC set has MORE THAN 6 HOURS of video tutorials and will teach you how to start doing great Character Animation with LightWave 5.6! PLUS: An interview with Stephen Bailey, LightWave renderings, objects, and textures!
Available NOW for ONLY $49.95!

TutorVision3-D CD Volume 3
Almost 6 and a half HOURS of video tutorials for LightWave and Aura! Topics covered include advanced motion control, beginner & intermediate compositing techniques, and numerous tips for using SkyTracer, Steamer, HyperVoxels, and ParticleStorm effectively! Plus MUCH MORE! This CD covers a bit of EVERYTHING and is an INCREDIBLE VALUE!
Available NOW for ONLY $49.95!

TutorVision3-D CD Volume 4
The ULTIMATE Beginners Guide to LightWave! This CD demonstrates many concepts and techniques that all LightWave users need to know before they can get THE MOST out of LightWave...Frequently-Asked-Questions are answered in easy-to-understand video and text tutorials! If you are learning LightWave 5.6, then you NEED this CD!
Available NOW for ONLY $49.95!

The LightWave Mailing List Search Engine CD
How would you like to have THOUSANDS of other LightWave users waiting to answer your questions? This amazing CD allows you to tap into the archived knowledge of the LightWave Mailing List with an advanced Search Engine that quickly searches through nearly 87,000 messages to find the answers and tips that you need for your LightWave work! Messages are from May 1997 to December 1999.
Available NOW for ONLY $19.95!

Tutorials for LightWave 6.0 COMING SOON!

So how do you get your FREE Sampler CD?

Just get your address to us via e-mail, phone, or use the special form on our web-site, and we'll mail the Sampler CD to you right away, with no obligation to buy anything!

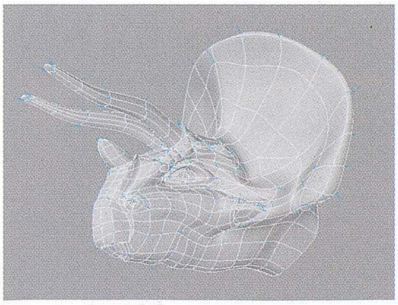
SAMPLER DISC

Quantities Limited - ORDER NOW!

Event Horizon Studios
Phone: 518-756-6850
FAX: 518-756-6836
E-Mail: info@tv3d.com
Web-Site: http://www.tv3d.com

Back to the Basics

by Mike Reiser



One of the things that seem to hinder LightWave modelers, beginners in particular, is lack of preparation. More often than not, the question of where to begin comes up. If you ask someone who is having difficulty how far he or she has gotten, the answer is usually something like, "Well, I have this spline cage that looks like a blob." I ask them first what their idea is, and what exactly it is that they want to model. A lot of the time, to my surprise, people may not have a clear-cut goal of what they are trying to achieve. They are just so excited about getting on the computer and firing up LightWave, that all else falls by the wayside.

Having the software and the new computer does not mean that you should forget the basics of preparing a project. If you have the ability to simply open Modeler and create a wonderful creature from scratch, you're a better modeler than most. The rest of us need to prepare a little more carefully. Plan! Plan! Plan! Stop where you are, turn off the computer and grab a pencil and paper. In the real world, one cannot simply sit down with the client and model in LightWave so effectively that you can discuss ideas efficiently. More realistically, sketches, paintings and maquettes are made prior to any computer even being turned on. Don't get me wrong: I am not saying that you have to be Michelangelo, but get the ideas down on paper first. If you like to sculpt, try putting your idea in the form of a maquette. Sketching and sculpting may save you hours of LightWave modeling if your client says, "Sorry, not what I had in mind." It doesn't make a difference if your idea is a sketch on a cocktail napkin or the most incredible maquette that was ever sculpted. The point is to plan first, then go to Modeler. You will be amazed at how much better and more efficient your modeling can become.

Let me get off my soapbox and take you through an example. For this tutorial, I commissioned myself to make a model of a Triceratops. Ever since "Walking with Dinosaurs" I have had this fascination for prehistoric creatures. Now, everybody knows what a Triceratops looks like, right? Three horns, that big flat thing in his head, four legs and a tail. OK, but how far apart are his eyes? Where do those horns come out of his head? These are the kind of answers that come after a little research and preparation. So I did just that. I looked at any and every image that I could find. I spent more time in the kid's section at the bookstore than I care to tell. Below is a sketch that I came up with that would act as a guide for modeling.

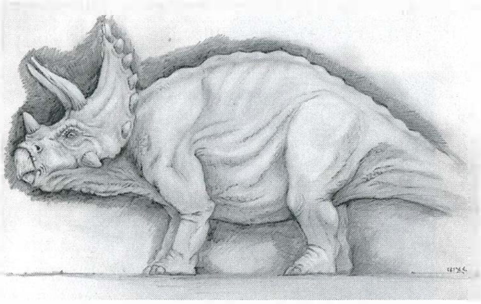


Figure 1: Sketch of triceratops.

After a while, and some LightWave models that did not turn out so hot, I realized that I needed more than a sketch. I needed to see and feel the third dimension of what I was doing. This is when I decided to make a maquette of the Triceratops. This is really easy to do, and very inexpensive. Go to your local art supply store and grab some modeling clay. Get at least four or five pounds. Go home and raid the closet for coat hangers. (The wire kind, not the wood!) Find a piece of wood to use a base. Start cutting and bending until you get a wire version of your creature. Electrical tape works great to hold it all together. I like to nail down the feet so that the wire frame is secure.

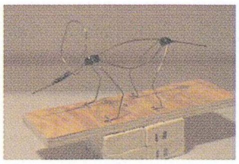


Figure 2: Wire armature.

If you don't want to use clay for the bulk of your maquette, try forming a wad of tinfoil around the armature instead. This can be a money-saver if you are short on clay.

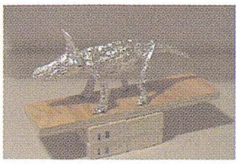


Figure 3: Tinfoil added to armature.

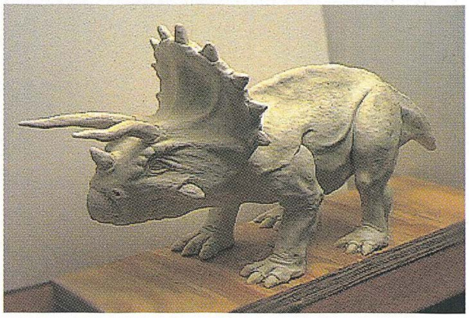


Figure 4: Clay maquette.

This is the actual progression of my Triceratops maquette. I started out with a coat hanger and tinfoil, then added modeling clay.

Above is my final maquette that I ended up using. It took about six hours to complete, so my life was not spent sculpting. Don't be too worried about the little details. You can make scales, small wrinkles and other tiny details later, in Modeler or with a bump map.

Use the maquette as a guide for spatial perception. The sculpting process helps force you into thinking about your idea in three dimensions. I find it easier to hold, flip and rotate a physical model than to do so using line drawings alone. Also, having the maquette right next to you while you model gives you the ultimate reference!

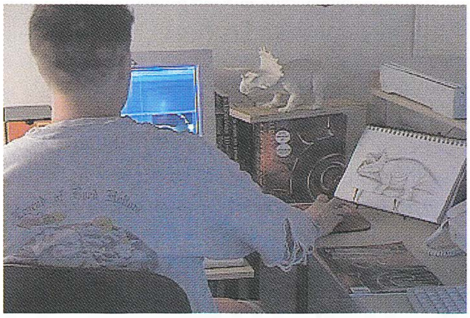


Figure 5: Maquette and drawings as aids.

At this point, you are much better prepared to start modeling in LightWave. There should be very few unanswered questions on how you want your model to look. The client should also have a very good idea of where you are headed. It's far less likely that you will have to remodel anything. So, with the client's approval, let's turn on the computer.

That lump of clay can help you out in more ways than one. In addition to having

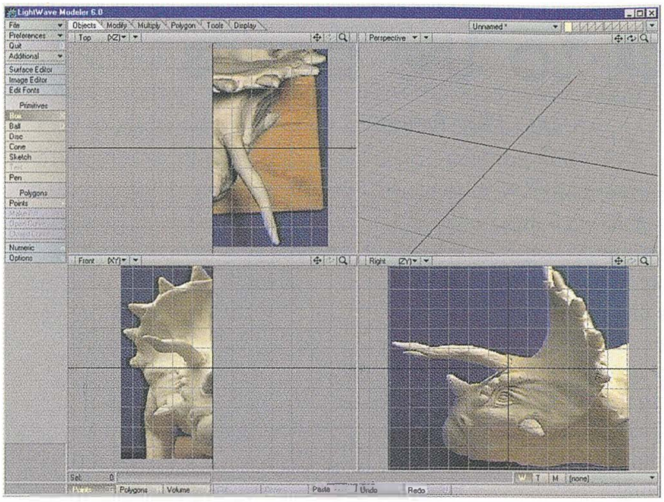


Figure 6: Maquette images as backdrops.

the maquette next to you for quick reference, place it into the background of Modeler. What I did was take digital images of the Triceratops and place them as templates in the background.

In order to place the image in the background, go to the Display tab and click on Backdrop. Simply load each image into its respective orthogonal view. (If you are only going to have one view, make it the right or left. This view gives the most information.) Some things to keep in mind are the Backdrop image position and size. In order to make the images line up correctly (or close, anyway) pick a midline structure and place it at 0,0,0. I chose the tip of the nose horn, as you can see. Use the position spinner in Backdrop to move the image into position.

Unless you are very careful when digitizing the images, the size of the maquette may very slightly from one image to the next. To remedy this, use the size tool in the backdrop menu to change the size of the images so that they are all roughly the same. Use the grid lines to line up structures in one view to the same structures in another view.

You will have noticed that I have used only half images. This is a personal choice, simply because I prefer to model in half and then use the mirror and merge points tools to complete the model. LightWave now has a symmetry option. It's very handy, and I suggest using it if you have not tried it. However, I still like to model in halves. I will model only the head in this tutorial, and I will discuss various tools and demonstrate some LightWave modeling techniques that I think are very useful.

There are various ways to begin the modeling process in LightWave. With complicated heads such as this one, I prefer to start out with a planar set of polygons, then add depth by pulling the points into a rough approximation of the head.

Drag out a box in the right viewport that has enough detail so that you can move the edge points to roughly outline the Triceratops head, as shown in Figure 7. Don't worry about the crown; we will add that later.

Move the points to make a rough approximation of the head as shown in Figure 8.

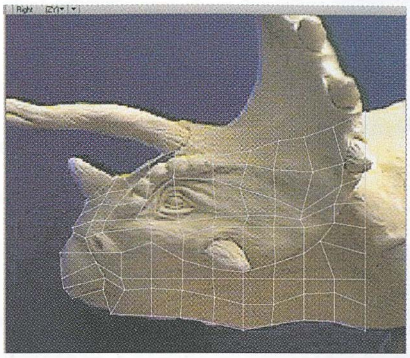


Figure 8: Points repositioned.

Your cage does not need to look exactly like this, but try to move the points so that they come close to outlining the major features that will eventually have three dimensions. The model is still planar at this stage; the next step is to turn it into a 3D object. Go to the Multiply panel and use the Smooth Shift tool to add depth to the planar object so it is as wide as the Triceratops beak. This is the first level of the extrusion process.

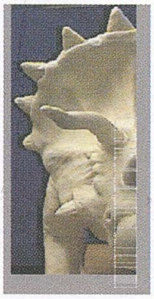


Figure 9: Extrude.

Smooth Shift is probably my favorite and most used LightWave modeling tool. It is very much like Bevel, but it treats the selected group of polygons as one shift, hence the name. I suggest you use it often. It can be a great time saver when you need to add detail to certain parts of a model. The Smooth Shift method that works well for me, and provides the most control with the least amount of headaches, is to select the polygons and Smooth Shift with an offset of zero. The new polygons remain in the same position so the model is not completely

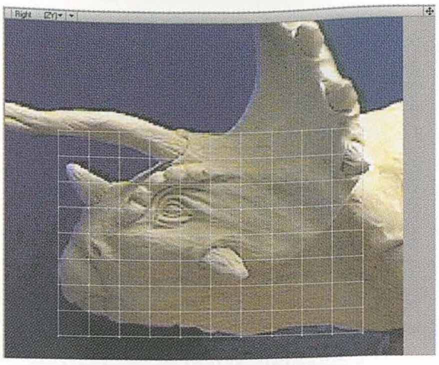


Figure 7: Box outlines head.

distorted. I then use the move, rotate, drag and stretch tools to adjust the polygons, as shown in Figure 10.

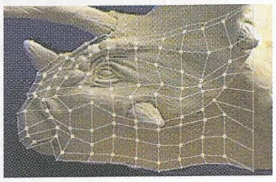


Figure 10: Points adjusted to match head.

Go to the Modify panel. Using a combination of the maquette, sketches, and backdrop images, use the Drag tool to adjust points so that they begin to take the shape of the Triceratops head, Figure 11. This is the point when the maquette is invaluable. You should be constantly checking your model and comparing it to the maquette. Take one section at a time; don't do everything at once.

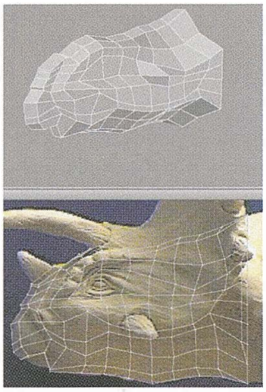


Figure 11: Drag points.

Look at the mouth first. You will see that I left out the bottom jaw when I made my original outline. Let's put that in now. Select the polygon that will be the lower jaw, as shown in Figure 12.

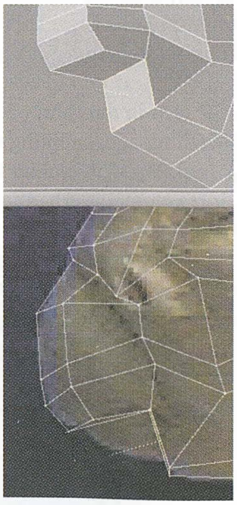


Figure 12: Select points.

Use the Smooth Shift tool and pull

out the lower jaw. You will want to do this twice, so as to give yourself enough polygons to shape the mouth. You should end up with something like Figure 13, below. Delete any extra polygons that you may have created on the back side of the half-head during the Smooth Shift.

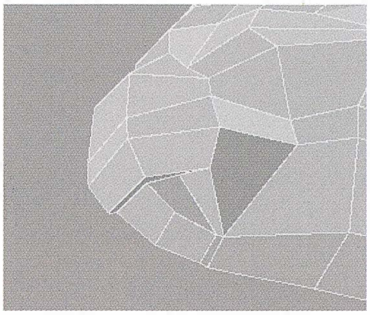


Figure 13: Lower jaw points.

New to LightWave, and a great addition, is the ability to add a 'weight' to a point while using the SubPatch option. (The old MetaNURBS.) To use this, press the Tab key, which is the hotkey for SubPatch. Note that your model is now smoothed out. If you have any polygons that are not 3- or 4-point polygons, the result will not be pretty. If need be, correct any polygons so that they meet this qualification.

Highlight the "W" button in the lower right hand corner of the Modeler interface to select the SubPatch Weight option. Note that the "Weight" and "Weight Brush" tools are now active in the Modify panel.

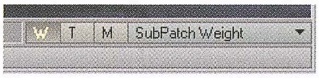


Figure 14: Select Weight option.

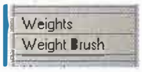


Figure 15: Modify panel Weight features.

Hold down the left mouse button over a point and drag. You should see the spline lines change shape around the point, either sharpen or relax, depending on the direction you drag the mouse.

But before you start changing any weight values, let's add some more detail to the mouth. To do this, we are going to look at another new tool, BandSaw. Select two polygons near the snout adjacent to the mouth. Go to the Additional button under the Objects Tab. Find the plug-in called BandSaw. With the two polygons selected, use the default values that BandSaw provides and click OK. In the direction of the long axis of the polygons you selected, a row of polygons around the object will be split. This is a very cool way to add detail to selected sections of the model.

Use the BandSaw to add detail to the snout of the Triceratops. Use the drag tool as well as a weight map to give the snout a little more detail. Select the polygons where the nostril would be, and use Smooth Shift a number of times to create the nostril. All the while, you should be adding detail by adjusting individual points.

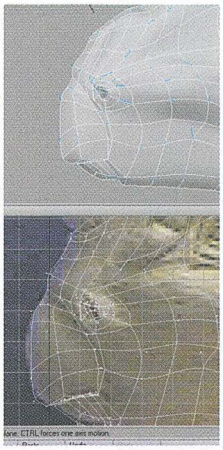


Figure 16: Snout details.

Having done the snout, let's move on to something a little tougher to pull off, the eye. This will involve a number of techniques, including splitting polygons and adding points. For the longest time I was troubled by why my models had that Smooth Shift look. (Well, for one, that's all I was using!) Then one day I noticed that if I split the polygons where I needed, I could control the model much more easily. Remember, you are not limited to what MetaForm or Smooth Shift gives you. As long as you remember to maintain the 3- or 4-point polygon rule, you will be OK.

Let's look at how to add detail to the eye of this beast. The ridge above the eye is the easy part. Simply select the polygons above the eye that are roughly in the same position as the ridge in the backdrop images, and use the Bevel tool. You will have to fine-tune the end result.

The next step is to select the polygons that make up the area of the eye itself. This part will require splitting polygons. Because we are going to be in tight, we need to hide some of the surrounding model so we can see more clearly. With the eye polygons selected, go to the Display Tab and click the Hide Unselected option. All that should remain are the dozen or so polygons that you selected for the eye.

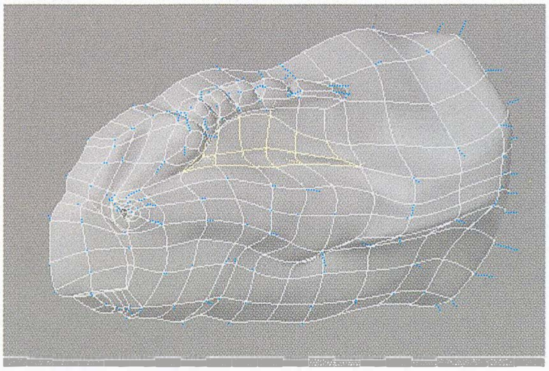


Figure 17: Polygons around eye selected for splitting.

Smooth Shift the eye polygons a few times, one inside of the other. Stretch the polygons each time, so they are just smaller than the surrounding set of points. Do this about 4 times. You may find that you need to add detail across an area that is a four-point polygon. You need to split this polygon into a 3-point. First, select the polygon you wish to split. Then go to point mode, and select the points you wish to have the split run through. Now, under the Polygon tab, select the Split tool. A new spline should now run between the points you selected. Adjust the points by using the drag tool, and unhide the rest of the model. You can repeat the Smooth Shift at different levels as many times as you need in order to get even more detail in the eye.

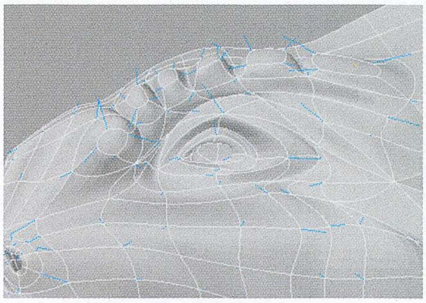


Figure 18: Eye details.

The little side horn is just a matter of Smooth Shifting a few times. Select the polygons that will make up the area just before the crown starts. Smooth Shift them a few times, each time adjusting the points. You should end up with something like Figure 19.

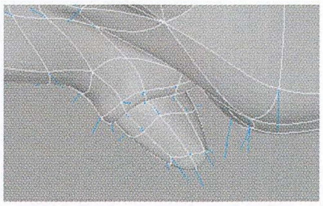


Figure 19: Side horn.

The crown for this guy is one of the easiest things to do. Select a number of polygons that would be the base of the crown, as shown in Figure 20, and Smooth Shift a few times, each time adjusting the points and polygons inbetween each shift.

Continued on page 18

The Creature Core Project

by Paul Davies

www.savagefrog.com info@savagefrog.com

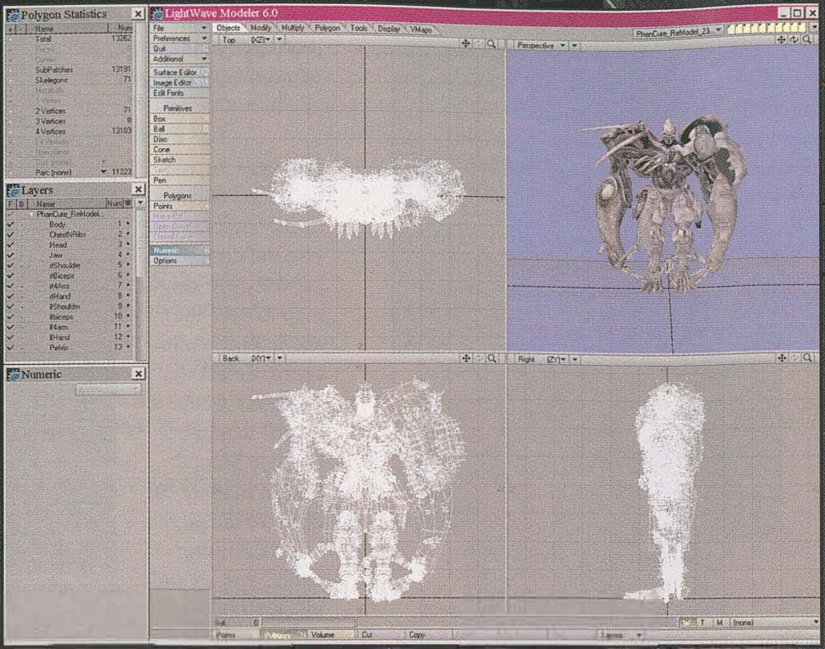
Savage Frog! Animation Studios' goal is to create high quality animation, primarily character animation. Recently, Ken Sullivan and I, two of the three founding members of Savage Frog!, realized that our reel was completely comprised of cartoon-oriented 3D animation. Proud as we were of the work, we felt that it only showed a portion of what we hoped to bring to life.

Two of the focuses we had yet to address were photorealism and creature animation. We decided to tackle both with The Creature Core Project. The idea was to make the unbelievable believable: an unidentified videographer captures footage of a monster stalking him in a deserted alley.

The Set

The alley in question is completely fictitious, and it was conjured from Ken's imagination based on reference footage from various cities and on site surveys from the streets of San Francisco. The set was constructed in a way we felt would be interesting to the eye, yet believable as someplace you might find in a big city. Everything in the set was modeled to scale and textured from reference photos shot from various locations. (You have to love digital cameras.)

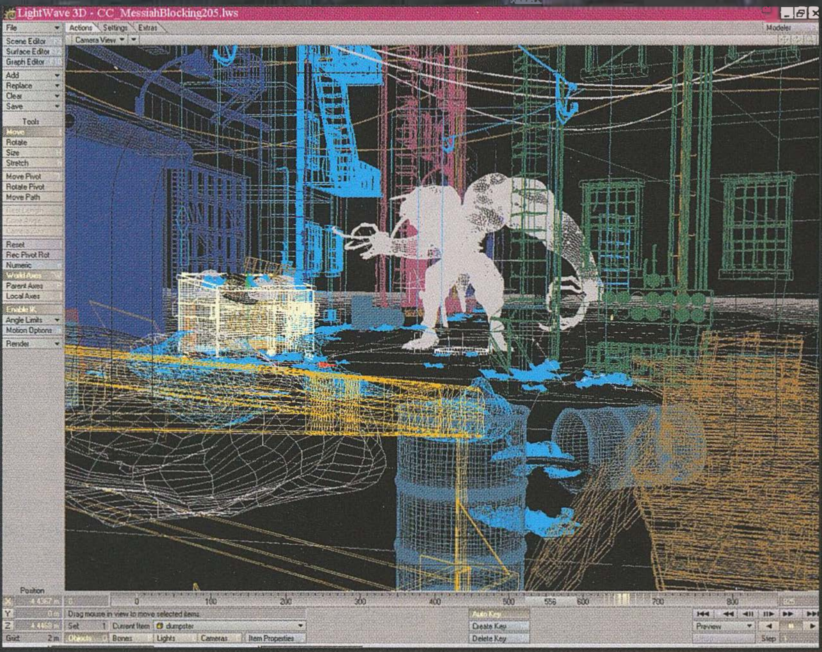
We wanted to keep the polygon count as low as possible. However, Ken found it necessary to model many of the imperfections and not to rely too much on bump maps. There were still an ample number of image maps used. There are 136 image maps in the alley, most of which are altered photographs; some are hand painted textures and customized, rendered DarkTree textures. Ken used a few L[6] procedural textures as well. The final scene weighs in at 218 objects with a total of 820,190 polygons. Ken also created a low-polygon version of just the key alley objects as reference for me to animate Phancure.


Polygon Statistics
LightWave Modeler 6.0
Layers
Numeric
Perspective
Top
Back
Right

Phancure was modeled entirely with L[6] SubPatch surfaces. The final render version was converted to polygons.

Ken endured numerous grueling critiques during dailies. As soon as he got an element to everyone's liking, we would point out something else that needed attention. In an attempt to dethrone me as king of the control freaks, Ken hand-animated bits of trash reacting to Phancure's violent activities. I think we are tied for the throne. In the end, Ken's attention to minute details really paid off. The Creature Core alley is truly amazing and quite convincing.

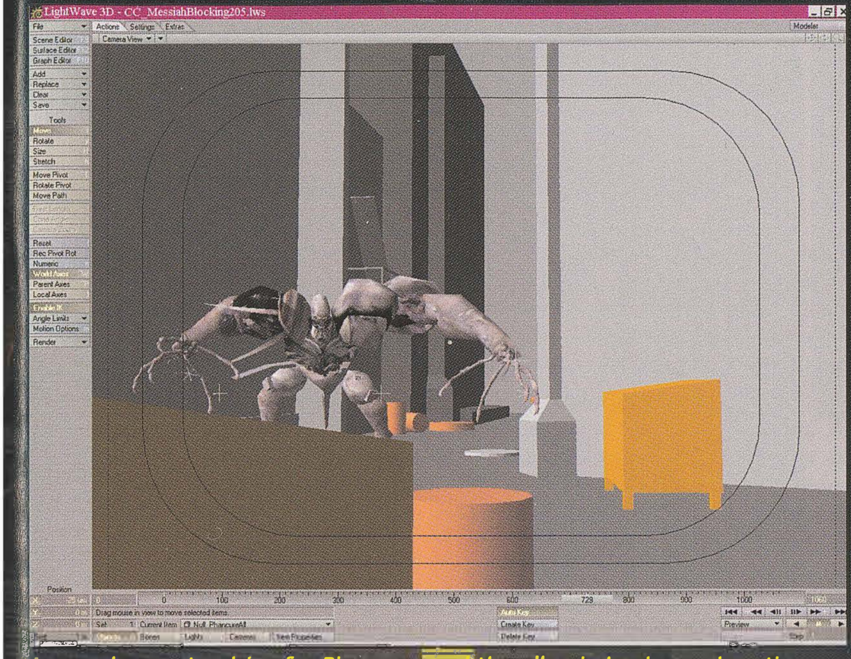
A prominent element in the scene is the 'steamy manhole' positioned in the middle of the alley. For the steam Ken went with Particle Storm 2 and L[6]'s HyperVoxels. To lessen the screen lag time in Particle Storm, Ken loaded the low-polygon alley elements and a low-polygon approximation of Phancure for positioning and collision purposes. Ken was able to get the look we wanted for the steam without too much tweaking; it just took some good pre-planning and a bit of what Ken calls luck. I refer to it as Ken's skill.


LightWave 3D - CC_MessiahBlocking205.lws
Actions
Settings
Extras
Camera View
Modeler
File
Scene Editor
Surface Editor
Graph Editor
Add
Replace
Clear
Save
Tools
Move
Rotate
Size
Stretch
Move Pivot
Rotate Pivot
Move Path
Reset
Rec Pivot Rot
Numeric
World Axes
Parent Axes
Local Axes
Enable IK
Angle Limits
Motion Options
Render
Position
Drag mouse in view to move selected items.
Current Item
dumpster
Objects
Bones
Lights
Cameras
Item Properties
Auto Key
Create Key
Delete Key
Preview
Step

This wireframe image of Phancure and the alley shows the high level of detail and effort used to create the models. If you look closely, you may be able to see the outlines of various signs on the buildings.

The Beast

Phancure, the monster to whom the videographer ultimately falls prey, is based on a sculpture by Yasushi Nirasawa. In order to accurately capture Nirasawa's magnificent design in 3D, I first scanned an image from Niraworks, a book filled with photos of Nirasawa's fantastically horrific models and sculptures. I then loaded the scan into L[6] Modeler as


LightWave 3D - CC_MessiahBlocking205.lws
Actions
Settings
Extras
Modeler
Camera View
Scene Editor
Surface Editor
Graph Editor
Add
Replace
Clear
Save
Tools
Move
Rotate
Size
Stretch
Move Pivot
Rotate Pivot
Move Path
Reset
Rec Pivot Rot
Numeric
World Axes
Parent Axes
Local Axes
Enable IK
Angle Limits
Motion Options
Render
Position
Drag mouse in view to move selected items.
Current Item
Null_PhancureAll
Bones
Lights
Cameras
Item Properties
Auto Key
Create Key
Delete Key
Preview
Step

Low polygon stand-ins for Phancure and the alley helped speed up the animation process.

a background and scaled it to [illegible] our estimation of Phancure's height. The scan allowed me to quickly build a crude approximation of the creature. My only concerns at this point were getting the proportions right, and identifying any issues that would affect the character setup and animation.

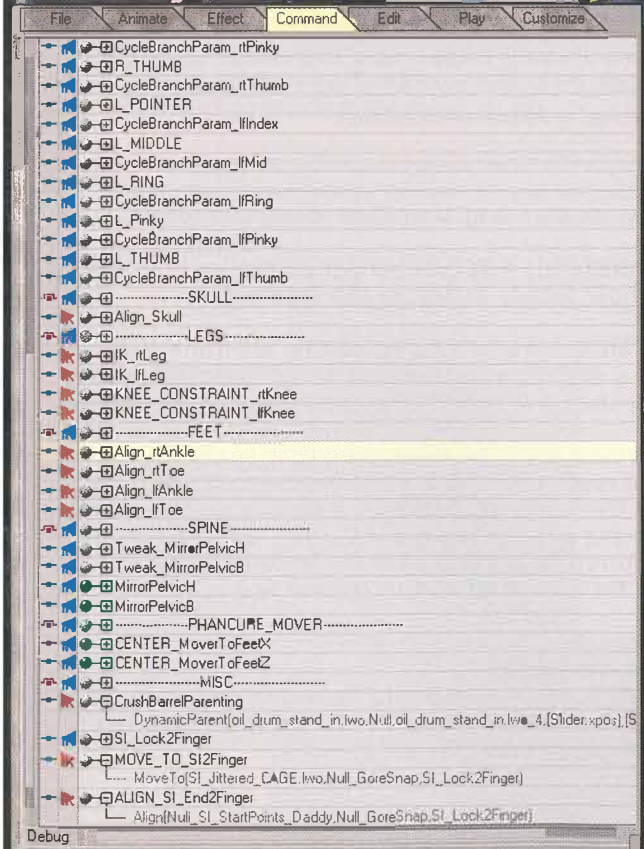
Jason Felix, another artist at Savage Frog!, modeled and textured the final high-resolution version of Phancure seen in The Creature Core Project. In order to get every little detail to match Nirasawa's model, Jason rescanned the original reference image at a higher resolution. There was still quite a bit of guesswork, because the only clear full body image of Phancure was a slightly angled front view. Jason used his best judgment (and a bit of nagging from the rest of us) to determine what Phancure's side and back should look like. The textures (45 in all) that comprise Phancure's rough hide are a combination of edited scan elements and hand-painted elements, all by Jason. The end result is a model we would be proud to show Nirasawa himself, if we knew how to contact him.

Character Setup

I must admit that I went through an unusual amount of difficulty setting Phancure up for animation. I created three completely different rigs for the creature, two in L[6] and one in Project:Messiah, in an attempt to ease the burden of realistically animating the hulking monstrosity.

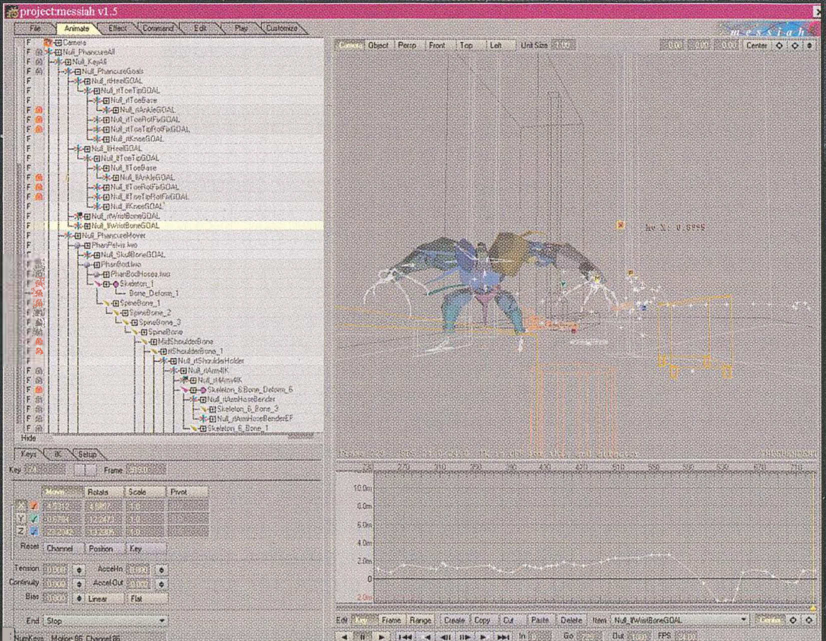
My first attempt was completely bone based. Although Phancure's exoskeleton is well suited for an object hierarchy, I initially chose to use a bone hierarchy. I have always found bone orientation to be intuitive, and have written LScripts for automatically creating IK chains complete with goals from bone chains. Using L[6]'s Skelegons and Weight Maps, it was relatively easy to setup a non-deforming bone hierarchy. I also used L[6]'s expressions on Phancure's spine and hips. Whenever I rotated the pelvis on Heading or Bank, the spine expressions automatically rotated the shoulders in the opposite direction. The hips also stayed halfway between Phancure's ankles via expressions. Not one to relinquish total control to the computer, I added special nulls which allowed me to tweak the effects of

Continued on page 25


File
Animate
Effect
Command
Edit
Play
Customize
Debug

```
CycleBranchParam_rtPinky
R_THUMB
CycleBranchParam_rtThumb
L_POINTER
CycleBranchParam_lfIndex
L_MIDDLE
CycleBranchParam_lfMid
L_RING
CycleBranchParam_lfRing
L_Pinky
CycleBranchParam_lfPinky
L_THUMB
CycleBranchParam_lfThumb
-----------------SKULL-------------------
Align_Skull
-----------------LEGS-------------------
IK_rtLeg
IK_lfLeg
KNEE_CONSTRAINT_rtKnee
KNEE_CONSTRAINT_lfKnee
-----------------FEET-------------------
Align_rtAnkle
Align_rtToe
Align_lfAnkle
Align_lfToe
-----------------SPINE-------------------
Tweak_MirrorPelvicH
Tweak_MirrorPelvicB
MirrorPelvicH
MirrorPelvicB
-----------------PHANCURE_MOVER-------------------
CENTER_MoverToFeetX
CENTER_MoverToFeetZ
-----------------MISC-------------------
CrushBarrelParenting
  └── DynamicParent(oil_drum_stand_in.lwo,Null,oil_drum_stand_in.lwo_4,[Slider:xpos],[S
SI_Lock2Finger
MOVE_TO_SI2Finger
  └── MoveTo(SI_Jittered_CAGE.lwo,Null_GoreSnap,SI_Lock2Finger)
ALIGN_SI_End2Finger
  └── Align(Null_SI_StartPoints_Daddy,Null_GoreSnap,SI_Lock2Finger)
```

Here are some of the expressions used to control Phancure in Project:Messiah.


project:messiah v1.5
File
Animate
Effect
Command
Edit
Play
Customize
messiah
Camera
Object
Persp
Front
Top
Left
Unit Size
Center
Keys
IK
Setup
Key
Frame
Move
Rotate
Scale
Pivot
Reset
Channel
Position
Key
Tension
Continuity
Bias
Accel-In
Accel-Out
Linear
Flat
End
Stop
NumKeys
Motion 86 Channel 86
Edit
Frame
Range
Create
Copy
Cut
Paste
Delete
Items
Null_lfWristBoneGOAL
Go
Out
FPS

All of Phancure's animation was created in Project:Messiah. This image shows some of the large hierarchy and numerous keys.

Continued from page 15

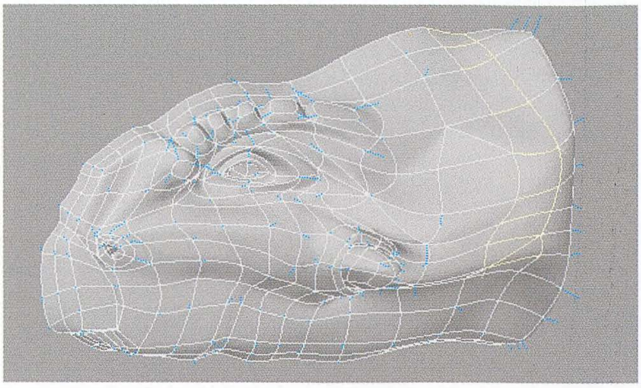


Figure 20: Crown polygons selected.

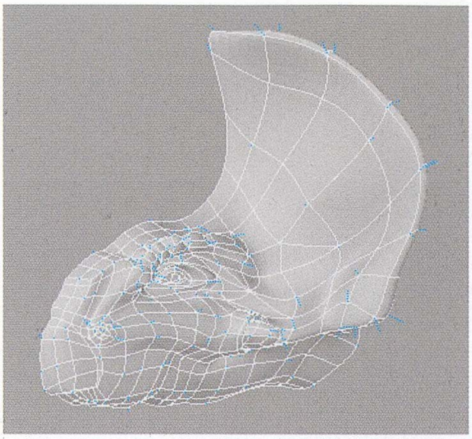


Figure 21: Crown polygons Smooth Shifted.

You should end up with something like Figure 21. Don't have too many Smooth Shifts for the crown. You want it to have wide sweeping lines, so use fewer shifts with space in between.

Now mirror and merge the points to make this guy whole. The final horns are, again, made using simple Smooth Shift technique.

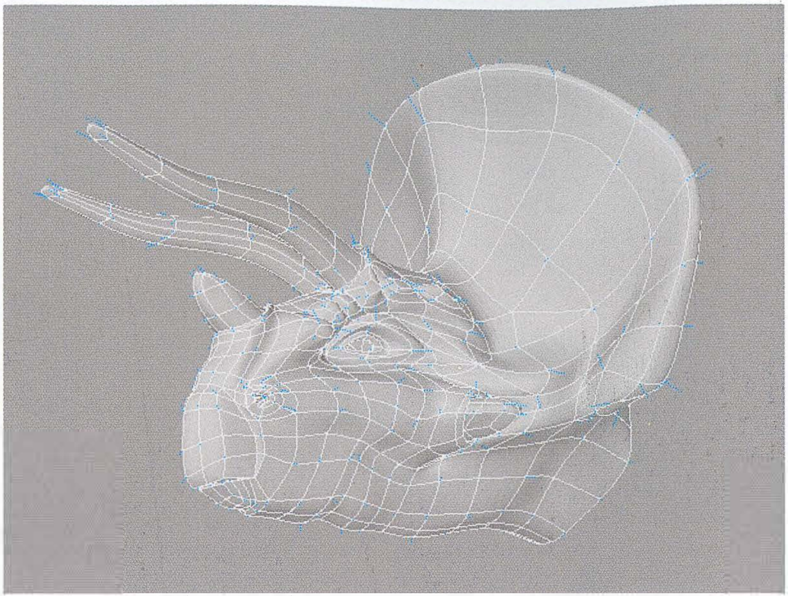


Figure 22: Final low-polygon Triceratops head.

Above is the final low-polygon version of the Triceratops head. As you can see, only a few tools and some simple techniques were used to make it. The body can be made using the same techniques.

The biggest help to completing this project was having planned out what I was going to make. I cannot urge you enough to plan. If planning means a simple drawing, great. If it means going full tilt and sculpting a maquette, even better. Just make sure that you do some kind of preparation. Above all else, have fun.

Michael "Doc" Reiser is a Radiology Resident in Pittsburgh, PA. He is co-founder of Pollywog Productions, a 3D animation studio specializing in character design, and medical and scientific visualization. He can be reached at ***MikeReiser@aol.com****.*

Continued from page 5

New and improved features of LightWave 6.1 include the following:

- Integrated particle system that includes interactive control over wind, particle effectors for turbulence and other effects, as well as gravity and precise controls over particle attributes such as, emission speed, birth rate, object collision (including subdivision objects) and particle resistance.

- Motion Designer soft-bodies dynamics engine integrated seamlessly into the interface, which enables simulations to be done interactively with other scene elements. The soft-bodies engine shares attributes with the particle system. For example, a wind effector could affect several particle emitters and a soft-body simulation at the same time.

- Character animation refinements for finer control of Skelegon creation, weight map enhancements and other tools for streamlining the process of character set up.

- New schematic view allows scene editing to be done in a nodal interface, which streamlines the process of hierarchy creation and simplifies scene overviews.

- StealthNET(tm) a TCP/IP render controller that lets users render locally or across the Internet.

- LSide, an integrated LScript editing environment that includes a debugger, editor and visual interface builder that simplifies and speeds the creation of LScripts.

- HyperVoxels(tm) improvements that enable HyperVoxels to read a particle's velocity and to deform the shape of the voxel accordingly, and to align the voxel shape to the path of the particle. New Sprites image mode allows images and animations to be attached to particles for advanced and incredibly fast particle effects.

Price and Availability
LightWave 6.1 is scheduled to be available late in Q3 2000. It will be offered free of charge to all registered owners of LightWave [6].

Character Shop

by Doug Kelly

Shave And A Haircut (Shave for short) is a standalone application and two LightWave plug-ins that, used together, enable you to add highly realistic hair, fur, grass and similar effects to LightWave objects. Shave does especially well at dynamics, making hair that moves realistically under the influence of gravity, wind, parent object motion, and collision detection. Shave is a very powerful program, and as such it requires a little work to learn. However, once you've mastered the basic steps of applying Shave hair to your objects, you'll quickly be able to produce very good-looking hair and fur. In fact, you would have to work pretty hard in the wrong direction to produce bad-looking hair. The default settings generate a tolerable first pass, and unless you ignore the evidence of your eyes, any changes you make should be for the better.

Shave was originally written for (and continues to work well with) LightWave 5.0 and 5.6. It is completely compatible with L[6]. The only significant sticking point for some users seems to be how to apply Shave to L[6] SubPatch objects, especially those with EndoMorph shape sets. Performing this operation successfully is not inherently difficult, but it does require a slightly deeper understanding of the whole modeling/Shave/layout process. This tutorial will spell out the details of this process, and along the way will introduce you to the major features and capabilities of Shave And A Haircut.

If you don't already own Shave And A Haircut, you can download a demo version from **www.joealter.com**.

Shave consists of a standalone application, a deformation plug-in, and a pixel filter plug-in. The application processes a LightWave model to create a hair file; the deformation plug-in applies the hair file to the rendered model; and the pixel filter plug-in renders the hairs, including the resulting shadows and other effects.

Because the actual rendering is a post-process effect, it is generally best to leave the Shave process to near the end of your animation workflow. Although Shave is a relatively fast renderer, as hair plug-ins go, it can still multiply your rendering times. You will do better to apply Shave after you have finished tweaking your models, lighting and animation. A change to any of these three factors will require you to repeat some (but usually not all) of the steps in the Shave process.

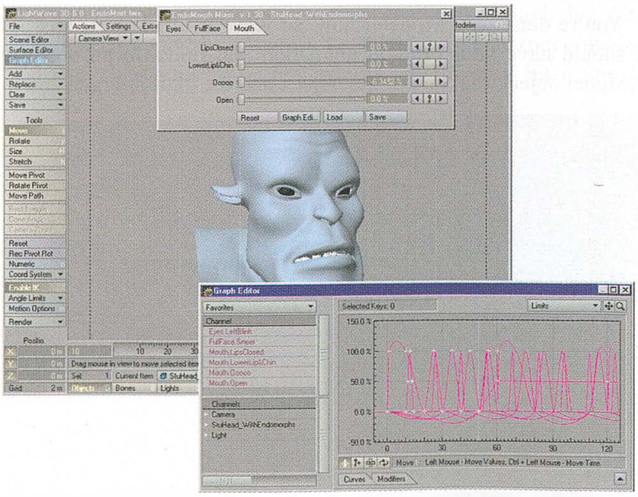


Figure 1: Layout with EndoMorph object loaded, EndoMorph Mixer active, and the Graph Editor displaying the looped actions.

For this tutorial, I simply loaded the EndoMorph head that is included with L[6], object file StuHead_WithEndoMorphs. I then animated the EndoMorph Mixer sliders to run through the basic shape set in 60 frames, and set all the graphs to loop. This provides an arbitrarily long animation that I can use to experiment with hair dynamics (motion and collision detection) without having to create additional keyframes. Figure 1 shows Layout with the object loaded, EndoMorph Mixer active, and the Graph Editor displaying the looped actions.

Once you have a completed EndoMorph animation, save the scene and clear Layout. Make a copy of the EndoMorph object. Load the copy into Modeler.

The next steps add collision surfaces to the copied object. Shave will use these surfaces to prevent rendered hairs from intersecting the object's skin. Collision surfaces are only necessary for areas that are likely to be contacted by long hairs. If you are applying Shave to a short-furred animal, if you are simulating grass, or if your character has a medium to short haircut, you can skip the collision surfaces.

As of this writing, Shave uses spheres for collision surfaces. This makes the calculation of collisions much faster, and it is easy to set up for most character objects. The calculations are based on the sphere's average radius, not the actual points and polygons, so you can use very simple spheres; the point/polygon count does not matter at all.

The simplest approach is to add a single collision sphere inside the character's head, as shown in Figure 2.

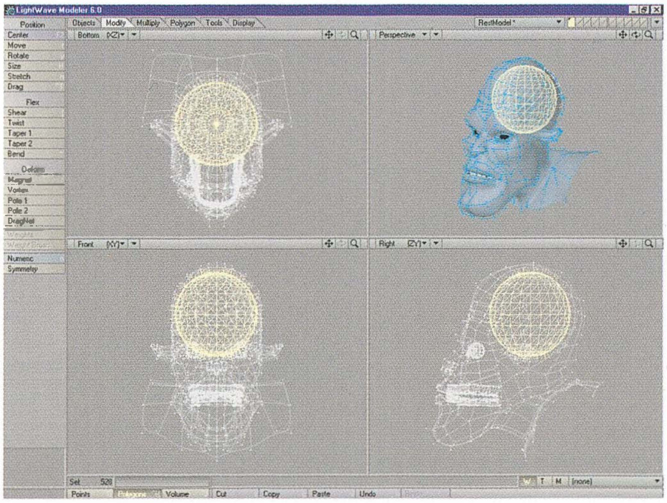


Figure 2: Single collision sphere added to object.

Note that the sphere is actually a little smaller than the character's head, so none of the sphere will be visible in the final rendering. Shave automatically increases the size of the collision sphere 7% during calculations, so you can safely hide the sphere inside the object without worrying about stray hairs glancing through your character's skin. There are more complex approaches to add and conceal more collision spheres, which I will mention later in this tutorial.

Change the surface of all the collision spheres to one named 'skull'. Only the skull surfaces will be used by Shave to calculate collisions.

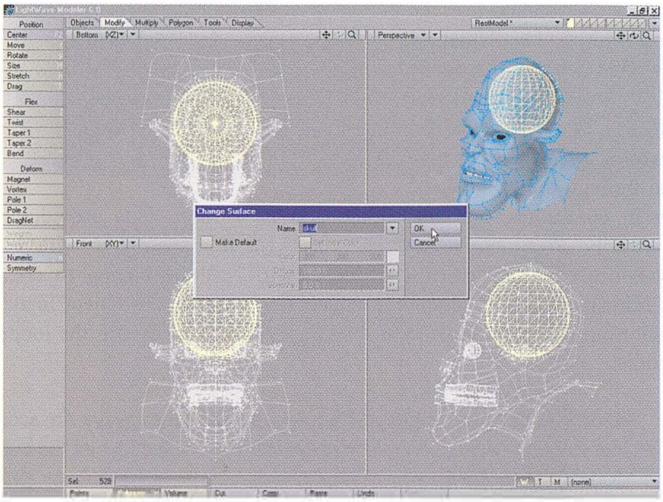


Figure 3: Changing collision sphere surfaces to 'skull'.

Deselect the collision spheres, and save the whole object as RestModel.lwo. This object should be identical to the original EndoMorph object in your animation, except for the additional collision sphere(s). All other surfaces should be intact. The next step will change that, so make sure you have RestModel saved in a safe location, and be careful not to save over it with the next stage.

Now you need to define the patches where you want hair to grow. Shave uses four predefined surface names to control hair growth: hair, eyebrow, eyelid, and beard. Note that these names must be all lowercase; if you misspell one, it won't work in Shave. If you're animating a creature with short, dense fur, the surface name 'beard' is probably closest to the effect you want.

When you have assigned hair surfaces to all the appropriate patches, save the whole object as SurfModel.lwo. Next, load SurfModel in Layout. Choose the Settings tab, and set SubPatch Level to 3 (or other, final-rendering level), as shown in Figure 5.

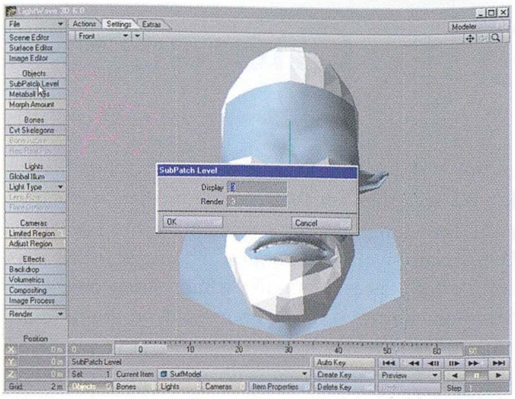


Figure 5: SurfModel loaded in Layout, and SubPatch Level set to 3.

From the Save menu, choose Save Transformed Object, and save the transformed object file as 3Growth.lwo. This object will have exactly the number of points and polygons as the final rendered SubPatch object; this match, like most morphing operations, is crucial to applying Shave to SubPatch objects.

One problem with using objects subdivided at the same level as the final rendering is that they have a lot of points and polygons, many more than you need for grooming hair in the Shave application. Having too much data can slow down screen redraw and suck up more RAM than necessary. You can avoid these problems by creating and working with a lower-polygon model in Shave, then transplanting the hair file onto the high-polygon model before rendering.

Set SubPatch Level to 0 or 1 (they both do the same thing). From the Save menu, choose Save Transformed Object and save the transformed object file as 0Growth.lwo.

Now you have a high-polygon model and a low-polygon model. The only remaining catch is that the L[6] object format includes some data structures that are incompatible with Shave And A Haircut's standalone application. This is an easy fix: Simply import both objects into Modeler, and export them in LightWave 5 object format.

Load 0Growth.lwo in Modeler. From the Additional menu, choose the Export_LW5 option. Save the object as 0GrowthL5.lwo. Close all objects.

Repeat the preceding step for 3Growth.lwo, the high-polygon model, saving it as 3GrowthL5.lwo.

You're done with Modeler now, so you can close it. At this point, you should have your original animation and EndoMorph object, the RestModel object, the SurfModel object, and the high-polygon and low-

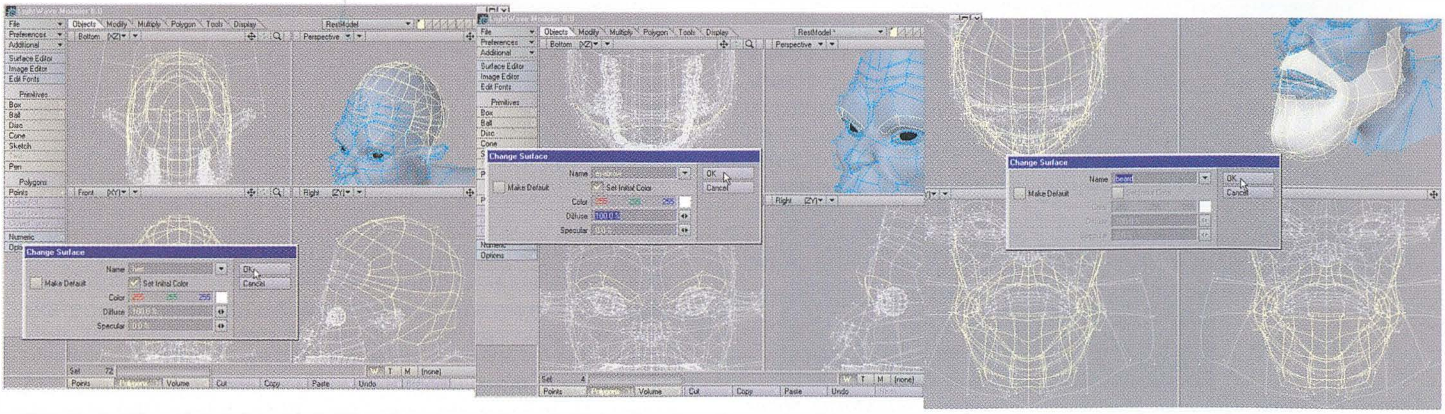
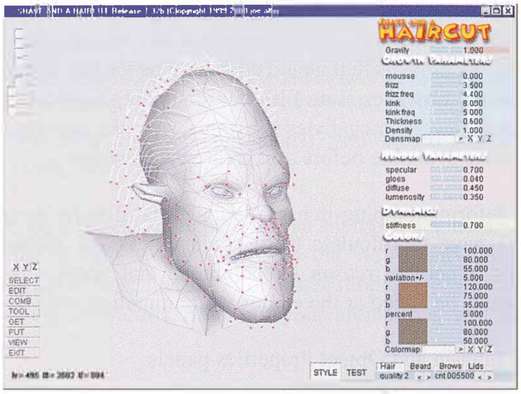


Figure 4: Changing selected SubPatches to surfaces hair, eyebrow and beard. This character doesn't appear to have eyelashes, so the 'eyelid' surface name is not used.

polygon L6 and L5-format objects. Don't worry about all this stuff you're creating; most of these files are intermediate materials, and you'll be able to delete them later. For now, keep them handy in case you want or need to repeat part of this process.

Fire up Shave And A Haircut. If you haven't used this program before, don't be intimidated. It has a lot of tools, but you only need to use a few at a time, and the default settings work very well for most characters. By the time you want to try something really exotic, you'll understand the more advanced tools you'll want to use.

First, load the low-polygon L5-format model, 0GrowthL5.lwo. In Shave, this is referred to as a growth model, because it is only used for growing hair. When you first load a model, Shave automatically performs a first-pass autogroom, applying the default settings for color, length and other parameters to the named hair surfaces as shown in Figure 6. Note that this relatively small number of hairs are only guide hairs; the actual rendering in Layout will be much denser. Having fewer guide hairs makes it easier for you to select hairs and to groom the model, and dramatically speeds up screen refresh.



Gravity 1.000
mousse 0.000
frizz 3.500
frizz freq 4.400
kink 8.000
kink freq 5.000
Thickness 0.600
Density 1.000
Densmap > X Y Z
specular 0.700
gloss 0.040
diffuse 0.450
lumenosity 0.350
stiffness 0.700
r 100.000
g 80.000
b 55.000
variation+/- 5.000
r 120.000
g 75.000
b 35.000
percent 5.000
r 100.000
g 80.000
b 50.000
Colormap > X Y Z
quality 2 < > cnt 005500 < >

Figure 6: *Autogroomed 0GrowthL5 low-polygon growth model.*

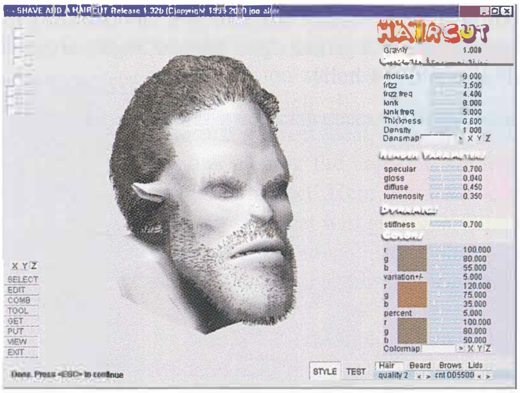
You can modify this grooming with a variety of tools. You can use the left-side menu options to change the hair length, to comb or translate hair in various directions, or to stand the hair on end, among other things. The right-hand sliders and controls change the rendered appearance of the hair and its dynamic behavior in LightWave Layout.

You should experiment with all the tools and controls in Shave And A Haircut, but you can significantly shorten your learning curve by reading the online manual and tutorials available through Joe Alter's website, listed previously. There isn't enough space here (nor would it be appropriate) to reprint the entire manual, so I'll only hit a few high points here.

Clicking on the Test Mode button enables you to try out dynamics. This mode is a good reason to use a low-polygon model, because your screen redraw will be much faster with fewer polygons and hairs. Click the Style button to return to Style mode without saving changes. Press ESC to save the current state of the hair. This can save you a longer dynamics run-up in Layout, especially with longer hair. If you use Test Mode to let the hair settle over the collision spheres according to gravity, the differences between the hair file you export and the hair's rest state in Layout will be much smaller.

Adjust Stiffness, and try Test Mode again to see how this parameter affects the dynamic movement of the hair.

You can choose the View>Draw Preview menu option to see a rendered preview of the hair, as shown in Figure 7. Press ESC to close the Preview.



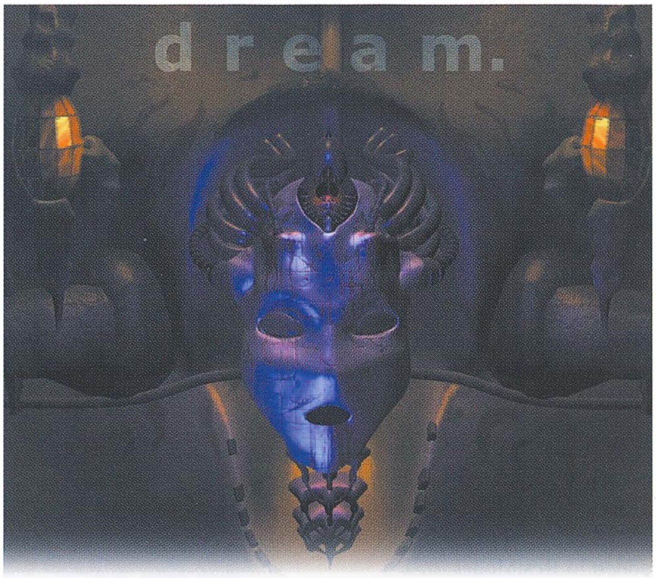
Done. Press <ESC> to continue

Gravity 1.000
mousse 9.000
frizz 3.500
frizz freq 4.400
kink 8.000
kink freq 5.000
Thickness 0.600
Density 1.000
Densmap > X Y Z
specular 0.700
gloss 0.040
diffuse 0.450
lumenosity 0.350
stiffness 0.700
r 100.000
g 80.000
b 55.000
variation+/- 5.000
r 120.000
g 75.000
b 35.000
percent 5.000
r 100.000
g 80.000
b 50.000
Colormap > X Y Z
quality 2 < > cnt 005500 < >

Figure 7: *Rendered preview of autogroomed model.*

When you are done experimenting (either because you're happy with your results or because you want to move on), choose the Put>Hair menu option and save the hair file as 0GrowthDefault.hair. This is just a backup file, in case something goes wrong with the next step. You can delete it later if you don't need it.

So far, you've applied and modified the hair of the low-polygon model. But you need a hair file that is compatible with the high-polygon count of the original SubPatch model. This is exactly what the Xplant feature was designed for. This enables you to import a model of a different resolution, but similar distribution and shape; and Shave will



3D Exchange
Animation
Training

Premier Authorized Training Center

NewTek

www.3dexchange.com • (510) 747-1540

automatically reassign the existing hair.

Choose the Get>Xplant Model menu option, and select the high-polygon 3GrowthL5.lwo object. SAAHC will automatically reapply the low-polygon model's hair grooming to the new, high-polygon model, as shown in Figure 8. Cool, huh?

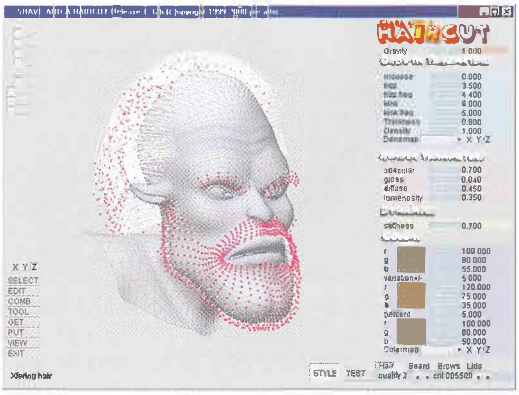

HAIRCUT
X Y Z
SELECT
EDIT
COMB
TOOL
GET
PUT
VIEW
EXIT
STYLE
TEST

Figure 8: *The Xplant operation transfers the low-polygon model's hair to the selected high-polygon object.*

This can be a huge time-saver, especially if you are working with models and hair densities that are close to your system's RAM and screen redraw limits. This approach enables you to get the most performance out of your computer.

If you like, make another Preview to see the difference between the low- and high-polygon versions of the same grooming. When you're done, choose the Put>Hair menu option again, and this time save the hair file with the name 3GrowthL5.hair.

At this point, you have the minimum two files you need to render hair in Layout: the RestModel object, and the 3GrowthL5.hair file. The other objects and hair files are intermediate files that you can discard once you finalize all settings for the rendering. However, I recommend that you archive them, just in case. In particular, backing up the SurfModel object can save you a lot of wasted time, because you can re-create growth models from it very quickly.

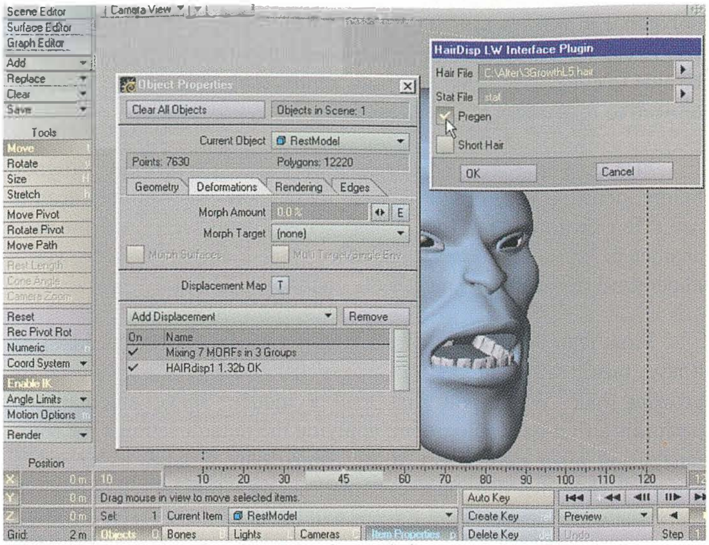
Open Layout. If you have enough RAM available, you can keep Shave and Layout open at the same time. You can make grooming changes to the .hair file, and save it out again, without making any changes to the scene in Layout. At rendering time, the plug-ins will automatically reload the updated hair file.

Load the scene containing your original animated EndoMorph character. Replace the EndoMorph object with RestModel.lwo. The only differences between these two objects are the collision sphere(s) you added to RestModel.

With RestModel selected, Open the Object Properties panel. In the Geometry tab, set both Display SubPatch Level and Render SubPatch Level to 3 (or whatever you used for the high-polygon version of Save Transformed Object, earlier; they must be the same, or an unpleasantness called Hair Explosion will result.)

Go to the Deformations panel and load the HAIRdisp1 plug-in. Open the plug-in. In the Hair File field, select the 3GrowthL5.hair file you saved from Shave. In the Stat File field, name a new stat file, "stat". Turn on Pregen. Your screen should look like Figure 9.

Dynamics in Shave are saved in stat files, one file per frame, that record which hairs are doing what. You create stat files by making a wireframe Preview in Layout. This calculates the stat files without a big rendering overhead, enabling you to conserve system resources and thereby to complete projects that would otherwise be too big or complex for your machine. However, stat files for complex, high-count hair movements can be several megabytes per frame, so make sure your hard drive has the space available before you create the Preview.


HairDisp LW Interface Plugin

Hair File C:\Alter\3GrowthL5.hair

Stat File stat

☑ Pregen

☐ Short Hair

Object Properties

Current Object RestModel

Points: 7630 Polygons: 12220

Morph Amount 0.0 %

Morph Target (none)

Mixing 7 MORFs in 3 Groups

HAIRdisp1 1.32b OK

Scene Editor
Surface Editor
Graph Editor
Add
Replace
Clear
Save
Tools
Move
Rotate
Size
Stretch
Move Pivot
Rotate Pivot
Move Path
Reset
Rec Pivot Rot
Numeric
Coord System
Enable IK
Angle Limits
Motion Options
Render
Position
Camera View
Clear All Objects
Objects in Scene: 1
Geometry
Deformations
Rendering
Edges
Displacement Map
Add Displacement
Remove
On
Name
OK
Cancel
Drag mouse in view to move selected items.
Auto Key
Create Key
Delete Key
Preview
Current Item
Objects
Bones
Lights
Cameras
Item Properties
Grid
Step

Figure 9: *Settings for the HAIRdisp1 deformation plug-in.*

Once the deformation plug-in has calculated stat files for an animation, you don't have to calculate them again unless you change the animation. This is another reason I recommend that Shave (like any other post-process) be added at the end of your animation workflow.

Close the plug-in and Object Properties panels.

If the dynamic motion of the hair is important, you should design your animations to include a run-up, a sequence of frames which will not be rendered but which are necessary to get everything moving. If you render a sequence without run-up, everything will start to move on the first frame from a standing start, as if it just snapped out of a time-warp. That's not usually the effect you want, so make sure you have at least 10 frames run-up before the beginning of your animation sequence. If necessary, use the Graph Editor to shift all your keyframes ten frames later, then render starting from frame 10 rather than frame 0. You'll need to run the Preview from frame 0, however, when you generate the stat files for dynamics.

In Preview Options, enable Virtual Memory, and choose Wireframe compression. Make a Preview of the entire animation, from frame 0 to the last frame you want to render. This will create a stat file for each frame. The stat files also enable you to use F9, or otherwise render any subset of images, and to have accurate dynamics for the rendering.

When the Preview is complete, End the Preview and then choose Free Preview to clear the wireframe preview from memory. Your content directory will now contain the stat files, one for each frame of the preview.

Open the RestModel properties panel again. Open the HAIRdisp1 plug-in, turn off Pregen, and close the plug-in and Object Properties panels.

In the Effects panel, add Pixel Filter hairRENDER plug-in. This plug-in has no options to adjust, so you can close the Effects panel.

Aim all the scene's lights (even the Distant and Area lights) as if they were all shadow-mapped spots to illuminate the hair to best effect. Shadow maps are most effective if they closely fit the illuminated subject. The shadow map has a finite size, and if most of it misses the subject that part of the map is wasted. If you are lighting with spotlights (which Joe Alter highly recommends), you should aim and size their cones to look like Figure 10.

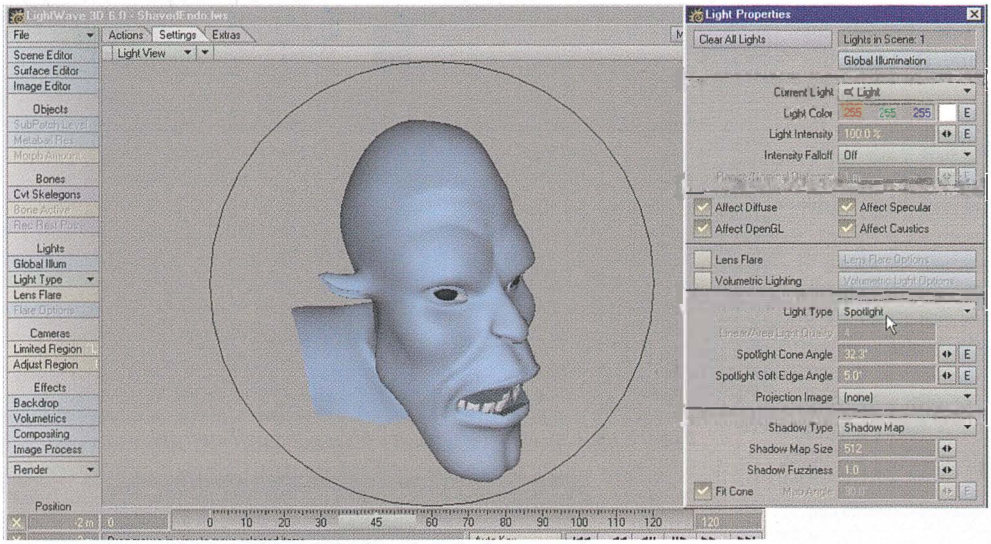

LightWave 3D 6.0 - ShavedEndo.lws
File
Actions
Settings
Extras
Light View
Scene Editor
Surface Editor
Image Editor
Objects
Bones
Cvt Skelegons
Lights
Global Illum
Light Type
Lens Flare
Cameras
Limited Region
Adjust Region
Effects
Backdrop
Volumetrics
Compositing
Image Process
Render
Position
Light Properties
Clear All Lights
Global Illumination
Lights in Scene: 1
Current Light: Light
Light Color: 255 255 255
Light Intensity: 100.0 %
Intensity Falloff: Off
☑ Affect Diffuse ☑ Affect Specular
☑ Affect OpenGL ☑ Affect Caustics
☐ Lens Flare
☐ Volumetric Lighting
Light Type: Spotlight
Spotlight Cone Angle: 32.3°
Spotlight Soft Edge Angle: 5.0°
Projection Image: (none)
Shadow Type: Shadow Map
Shadow Map Size: 512
Shadow Fuzziness: 1.0
☑ Fit Cone

***Figure 10:** Spotlight configured for best hair-lighting results.*

Set the Camera to render in one memory segment; and for best results, use at least Medium AntiAliasing, no Adaptive Sampling, with Motion Blur at least Normal. For faster rendering, you can choose no AntiAliasing or Motion Blur, but your animations will have a lot of chatter.

Change Render Options to display rendered images in the Image Viewer. Select a frame, and press F9 to render the current frame. Your results should look like one of the images in Figure 11. Try another frame, where the EndoMorph settings are markedly different. Note that the hair, especially the beard, precisely follows the deformation of the EndoMorph shape weighting. This demonstrates how well Shave works with L[6].

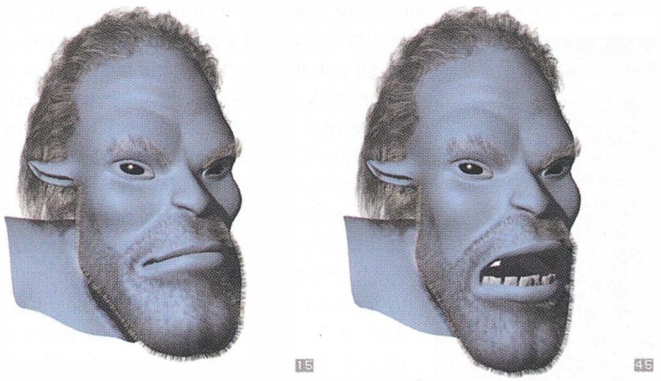


***Figure 11:** Frames 15 and 45 from a test animation, using auto-groomed hair.*

Rendering with the hairRENDER pixel filter plug-in active can multiply your rendering time. You should make sure that all your other animation, lighting and camera settings are complete before you start experimenting with the hairRENDER plug-in. If you do need to change any other settings, I suggest you temporarily disable the hairRENDER plug-in to speed up your test renders. The rest of your scene will render normally; only the hair will be missing. Once you have all the other parameters nailed down, you can enable the hairRENDER plug-in again.

If you have a character with very long hair, you may want to add more collision spheres to prevent hair from passing through the object. If the object deforms a great deal, you may also have trouble keeping the collision spheres hidden inside the object. Here are a few extra steps you can take that will give you full, animated control over your collision spheres, and will keep them invisible from the camera.

While RestModel is still loaded in Modeler, add more collision spheres. Surface them as 'skull', as before. Add them wherever you anticipate problems. For example, in Figure 11 you may have noticed a few stray hairs penetrating the character's right ear. I decided to eliminate that problem, and any unseen problems at the back of the head, neck and shoulders, by adding collision spheres as shown in Figure 12.

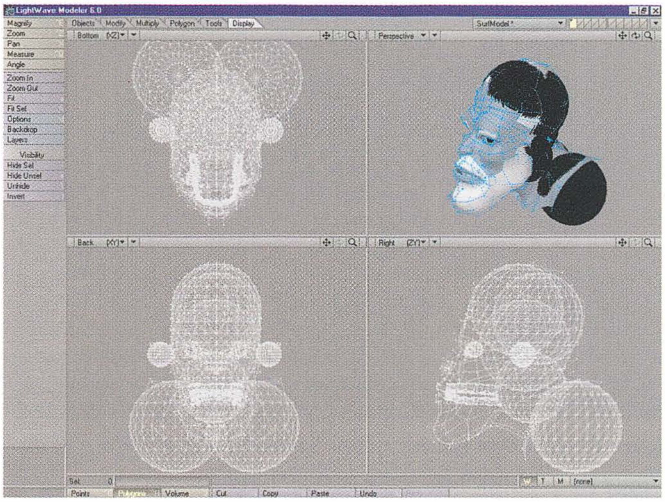

LightWave Modeler 6.0
Objects
Modify
Multiply
Polygon
Tools
Display
Magnify
Zoom
Pan
Measure
Angle
Zoom In
Zoom Out
Fit
Fit Sel
Options
Backdrop
Layers
Visibility
Hide Sel
Hide Unsel
Unhide
Invert
Bottom
Perspective
Back
Right
Points
Polygons
Volume
Cut
Copy
Paste
Undo

***Figure 12:**Plenty of collision spheres here!*

Once you're done adding spheres, select them all. Use the numeric requester to move the collision spheres a precise number of units in one direction, well clear of the rest of the object. Make a note of how far you move the spheres; you'll need that number later. You need to have enough room to apply bones with Limited Range, so the bones only affect the collision spheres and not the rest of the object. Figure 13 shows my approach; yours may vary.

Follow the rest of the process as before. After you replace the original EndoMorph object with RestModel, you'll have all those collision spheres hanging off the back. Add bones to the RestModel, and set the Limited Range, Rest Position, Rest Length, and Rest Rotation to nail down the collision spheres, as shown in Figure 14. Set a key for the bones at frame 0.

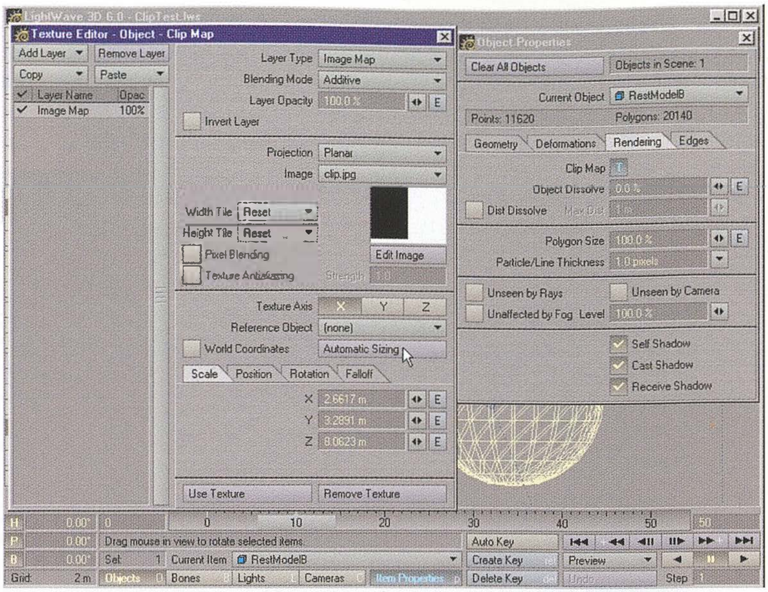
Now let's hide the collision spheres. We can't just make the skull surface 100% transparent, because there's a quirk (some say a bug) in L[6]'s renderer that makes perfectly transparent objects still show up in post-processing filters and buffers. Instead, you can apply a clip map to completely hide the spheres from the camera.

Select RestModel and open the Properties panel. In the Rendering tab, click on the T-for-Texture button labeled Clip Map. Apply the black-and-white clip.jpg map that's included with the Shave download, or make your own. It doesn't need to be fancy; it just needs to hide the spheres while leaving the rest of the object visible. Your panel settings should look like Figure 15. Close the panels.

Go to frame 1. Move the bones in precisely the reverse direction and distance you moved the collision spheres in Modeler. Set a key for the bones at frame 1. Your results should look like Figure 16.

You now have completely invisible collision spheres that are sized and positioned to repel hair from most of the character's surfaces. You can repeat the rest of the process as described earlier.

The difference is visible between Figure 11, with relatively short hair that still manages to interpenetrate the ear, and the lead image of this tutorial, with long, wildly flying hair that does not interpenetrate anywhere. Setting up collision spheres may seem like more work up front, but it surely saves lots of mistakes later.



Layer Type: Image Map
Blending Mode: Additive
Layer Opacity: 100.0 %
Projection: Planar
Image: clip.jpg
Width Tile: Reset
Height Tile: Reset
Texture Axis: X
Reference Object: (none)
X 2.6617 m
Y 3.2891 m
Z 8.0623 m
Current Object: RestModelB
Points: 11620
Polygons: 20140
Objects in Scene: 1
Clip Map: T
Object Dissolve: 0.0 %
Polygon Size: 100.0 %
Particle/Line Thickness: 1.0 pixels
☑ Self Shadow
☑ Cast Shadow
☑ Receive Shadow

Figure 15: *Clip map settings for hiding the collision spheres.*

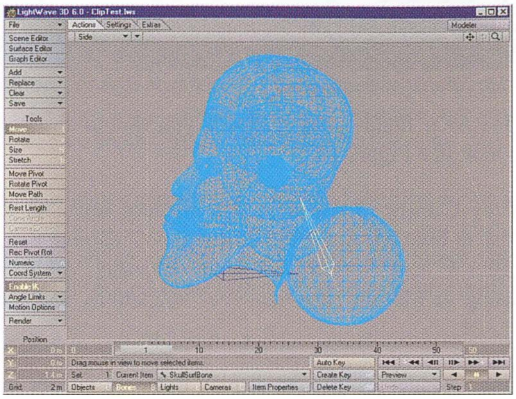


Figure 16: *Collision spheres repositioned inside object.*

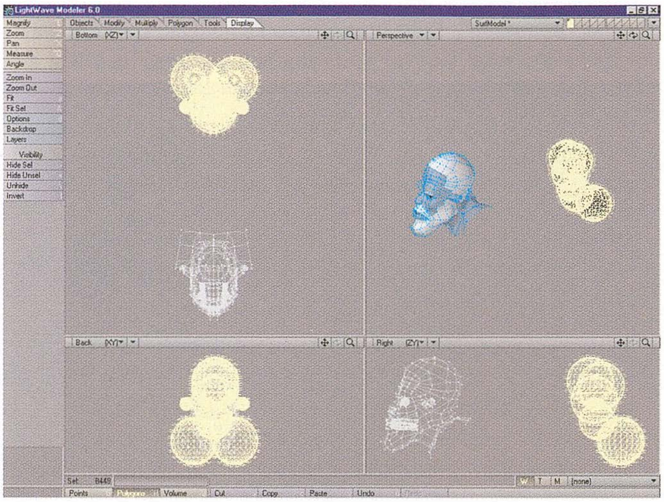
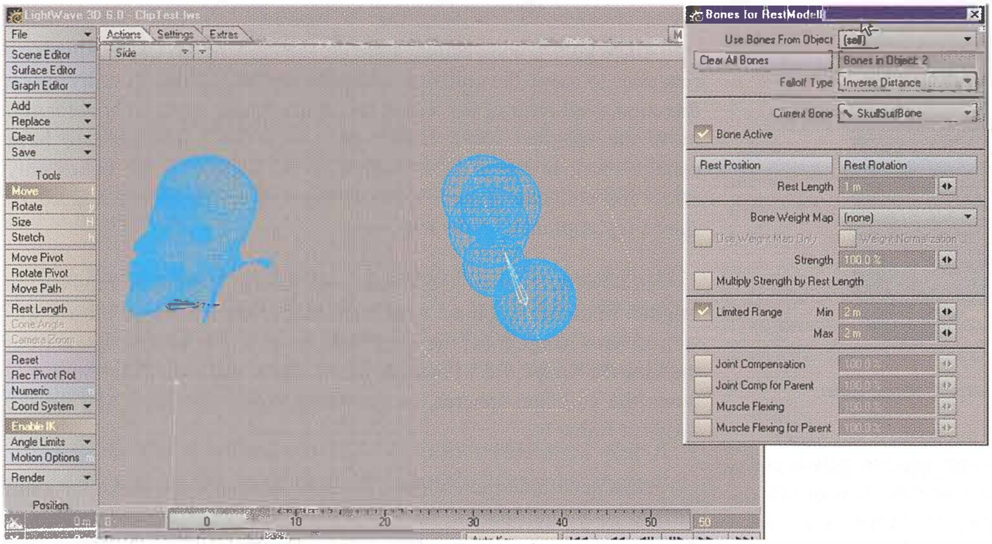


Figure 13: *Collision spheres moved 5.0 units clear of the rest of the object, providing room for bones to be applied.*



Use Bones From Object: (self)
Bones in Object: 2
Falloff Type: Inverse Distance
Current Bone: SkullSufBone
☑ Bone Active
Rest Length: 1 m
Bone Weight Map: (none)
Strength: 100.0 %
☐ Multiply Strength by Rest Length
☑ Limited Range Min 2 m Max 2 m
☐ Joint Compensation
☐ Joint Comp for Parent
☐ Muscle Flexing
☐ Muscle Flexing for Parent

Figure 14: *Bones in RestModel set to fix the shape of the collision spheres, and to insulate the rest of the object from the bones' influence.*

So how does Shave And A Haircut stack up against my five criteria for production tools? It provides many opportunities for incremental backups, many levels of undo, and a reasonable ability to recover work at any stage of the process. It allows you to adapt existing work to changed models, and to modify hair files right up to the final rendering, so it supports incremental detailing. The feedback is very rapid, as long as you work with reasonably-sized models and hair files, or you have a decent OpenGL accelerator card. The Preview renderings work well as documents for decision-makers to sign off. Finally, the process has a great deal of flexibility (as long as you pay attention to details), enabling you to mix-and-match hair files and models. All things considered, Shave And A Haircut makes a clean sweep of my criteria for professional product tools. Novices may find it slow going at first, but experienced LightWave users can get up to speed in no time.

You can email Doug Kelly at dakelly@earthlink.net.

Continued from page 17

the expressions.

Many people have complained about slow or no responsiveness while using LightWave's bones. Normally I find that keeping the Bounding Box Threshold low and viewing the bones in Bounding Box previews is sufficient for gauging a character performance. Unfortunately, bounding boxes and bones were not sufficient reference for Phancure. The combination of his unusual girth and regular interaction with items in the alley made it difficult to accurately gauge the performance in the usual manner. Eventually, I decided it was necessary to switch gears. I begrudgingly began the task of breaking the character into pieces for use in an object hierarchy. The task was made easier by L[6]'s multi-mesh feature: the ability to place pivot points and parent objects in L[6] Modeler and load all the parts as a single object with multiple hierarchically linked layers into L[6] Layout. I put my original Skelegons in a background layer and used them as a reference while placing pivot points for my object hierarchy. Matching the pivot points allowed me to maintain the articulation points from my bone setup. Since I used IK to animate the arms, legs and head in the boned version, I was able to use the same goals with their original motion to animate the new multi-mesh version of Phancure. Transferring the original motion allowed me to pick up where I left off with the bone animation. Unfortunately, the animation still had much room for improvement.

Timing is Everything

Timing is the most important aspect of good animation. I have seen truly expressive character poses lost in a blur of unidentifiable action because of timing problems. Our standard method of operation has been to animate, create preview, analyze, revise, render, then analyze and revise again. The Creature Core Project strained this methodology to the breaking point.

We decided early in the project that the entire piece should be seen from the victim's video camera with no cuts or transitions. Once I finished blocking out all of Phancure's action, the frame count had soared to 1200. To make things more manageable, I worked on the animation in sections based on Phancure's various actions throughout the scene. This only helped slightly, since most of Phancure's moves averaged 100 frames or more. By the time I started to refine the motion, I was spending more time compiling previews than animating. Eventually I began to skip the preview phase entirely, moving directly to network rendering my scenes for review. Although our farm is fast, I still had to wait to see the fruits of my labor. To make matters worse, I found it difficult to avoid fractional frame keys while scaling my keys. A new feature in L[6], fractional keys are a blessing when you want them and a curse when you don't.

As time marched on (and my animation did not), we decided it was time to film some reference footage. First, I surveyed Ken's digital alley set using a measuring tool I wrote in LScript. Then Ken grabbed his video camera and a tape measure, I grabbed some chalk and we headed to a nearby high school. Using the measurements I recorded, we recreated the alley on the school's blacktop using chalk outlines and trashcans. Next, Ken videotaped me doing my best monster impression as I moved through the makeshift set.

I planned to use the video as a timing template for my animation, eliminating the guesswork for the animation. As luck would have it, the video footage was not as helpful as we had hoped. Although the overall time was eerily close to 1200 frames, the movement of a 5'11" animator doesn't quite work when applied to a 12'5" monster. It was obvious that I would have to make considerable adjustments to achieve believable movement for Phancure.

Several people have raved about how fast Messiah's display is, so I decided to give it a try. Until now, we hadn't found it necessary to stray from the L[6] path. Switching to Messiah required a bit of work. I had to break Jason's layered version of Phancure into separate LW5.6 objects because Messiah was not compatible with L[6] at the time. I saved a 5.6 version of my base scene to load in Messiah. In L[6], Phancure's arms utilized Bank channel IK. Because Messiah's 2D IK is Pitch-based, I had to create a special null hierarchy for the arms. I then parented and realigned my arm objects to these nulls. I also got a crash course in Messiah expressions, where a considerable amount of Messiah's power lies. In addition to duplicating my L[6] spine and pelvis expressions, I added expressions to control Phancure's hands via sliders.

Despite the impressive refresh rate in Messiah, I could not get our scene to run at 30 fps on my DDR GeForce2. In retrospect, I'm sure we could have created an extremely low-polygon approximation of Phancure that would have been more refresh friendly in L[6] and Messiah. When you're hell-bent on completing a long project, it is often difficult to determine the best solution.

I happily settled for 15 frames per second and did not rely on previews for most of the remainder of the project! I did run into a slight problem due to this anti-preview methodology. At 15 frames per second, the animation looked smooth and natural; yet when rendered and played back at 30 frames per second, the motion seemed more linear and robotic. It is amazing how much of a difference every other frame makes.

Panic Cam

As previously mentioned, we wanted the camera to feel like someone shot this scene with a handheld camcorder. Ken was assigned the arduous task of animating the handycam effect. He studied the video footage we shot to see how the camera would act if the videographer were to be pursued by a twelve-foot monster. In addition to our reference video, Ken endured the horror of Blair Witch Project (again) to extract what he felt was effective and to avoid what was not so effective about their camera work.

Bring It All Together

To reduce render time and to allow a greater degree of control, the whole piece was broken down into several passes and composited with Digital Fusion. The render passes included background, volumetric lights, steam, character pass and character shadow pass. In addition to these layers, we applied grain, sepia tone and video noise to complete the illusion of reality. Though it is doubtful that anyone will believe Phancure to be a living, breathing creature loose in the streets, we do hope that viewers will have difficulty identifying this as a digital sequence. Instead, we hope viewers will have trouble determining how the scene was created. - *Paul Davies*

THE WILLIAM HOWARD TAFT COLLECTION:

7 EXECUTIVE POWERS FOR LIGHTWAVE 3D

Price : $199

Worley Laboratories
405 El Camino Real #121
Menlo Park CA 94025
650-322-7532
www.worley.com

Steve Worley has done it again. For sheer consistency in publishing software that does what it should, reliably and for a fair price, Worley should take home some sort of lifetime achievement award. Let's all hope he's not content to rest on his laurels.

Worley Laboratories' latest offering is another collection of specialized utility plug-ins that solve real-world production problems, or at least automate the stupid and boring parts of existing solutions. The Taft Collection is a set of seven plug-ins, designed to perform such useful tasks as gluing image maps to matched objects, stringing flexible hoses, and creating automatic gunfire and projectiles. This packs *lots* of fun into a small, affordable package.

The documentation is what we've come to expect from Worley: functional, well-organized, complete, and written with an offbeat sense of humor that makes the 68-page manual an entertaining read. You never know what obscure fact or outrageous fiction will come next. Even the blank page at the end, usually reserved for the user's notes, is retitled to a more ambitious purpose. (I won't give away the joke; buy the product, if you must know.)

39 demo scenes, 46 objects and 13 images are included on a second "Taft Extras" diskette. These demos are a very nice complement to the manual, and enable users to get up to speed quickly.

In addition to the publisher's direct support, you can subscribe to the worley-labs listserv on eGroups to share experiences, tips and tricks with other Worley Laboratories customers. To sign up, go to: **http://www.egroups.com/group/worley-labs**

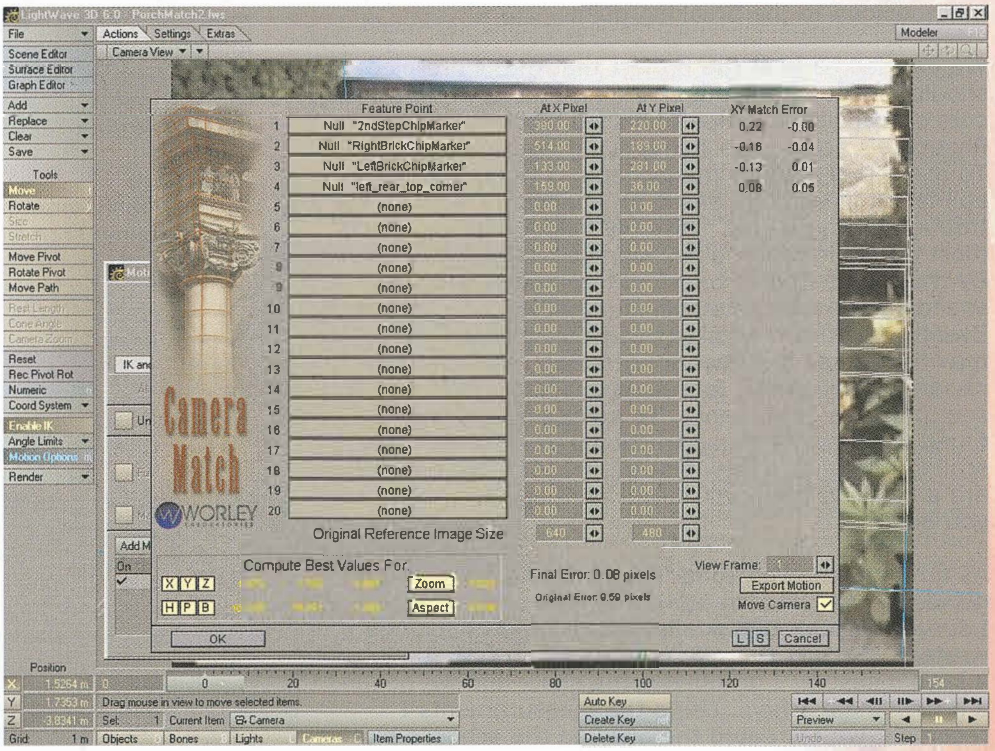
Like other Worley Laboratories products, the seven Taft plug-ins share several useful interface features. If you already use the Polk plug-ins, you'll be familiar with the Load, Save and Disable buttons, the Item Picker, and the general look and feel of the plug-in interfaces. To Worley's credit, all the controls do exactly what you'd expect them to (unlike some LW plug-ins, whose perpetrators should remain unappreciated).

Camera Match

Shortly before this plug-in collection shipped, I wrote a tutorial for manual move matching using LightWave for my book, *Digital Compositing In Depth*. I can attest that even a single-frame camera match is difficult and time-consuming, and a full six-axis move match can be nearly impossible. By using the Camera Match plug-in, you can significantly reduce the workload for both still matching and move matching.

Worley is very clear on the purpose of this plug-in: "A caveat: Camera Match is designed for use only with photographs, not animations. Animated camera tracking is significantly different and far more specialized." However, he also provides a concise half-page description of how to apply Camera Match to execute a successful move match. It's still not fun, but it's about 80 percent less work than a fully manual match move.

Camera Match requires a fair amount of up-front setup. If you didn't



	Feature Point	At X Pixel	At Y Pixel	XY Match Error	
1	Null "2ndStepChipMarker"	380.00	220.00	0.22	-0.00
2	Null "RightBrickChipMarker"	514.00	189.00	-0.16	-0.04
3	Null "LeftBrickChipMarker"	133.00	281.00	-0.13	0.01
4	Null "left_rear_top_corner"	159.00	36.00	0.08	0.05
5	(none)	0.00	0.00		
6	(none)	0.00	0.00		
7	(none)	0.00	0.00		
8	(none)	0.00	0.00		
9	(none)	0.00	0.00		
10	(none)	0.00	0.00		
11	(none)	0.00	0.00		
12	(none)	0.00	0.00		
13	(none)	0.00	0.00		
14	(none)	0.00	0.00		
15	(none)	0.00	0.00		
16	(none)	0.00	0.00		
17	(none)	0.00	0.00		
18	(none)	0.00	0.00		
19	(none)	0.00	0.00		
20	(none)	0.00	0.00		
	Original Reference Image Size	640	480		

Final Error: 0.08 pixels
Original Error: 0.59 pixels
View Frame: 1
Move Camera ☑

Camera Match
WORLEY
Compute Best Values For:
X Y Z
H P B
Zoom
Aspect
Export Motion
OK
L S
Cancel
File
Scene Editor
Surface Editor
Graph Editor
Add
Replace
Clear
Save
Tools
Move
Rotate
Move Pivot
Rotate Pivot
Move Path
Reset
Rec Pivot Rot
Numeric
Coord System
Enable IK
Angle Limits
Motion Options
Render
Actions
Settings
Extras
Camera View
Modeler
Position
Drag mouse in view to move selected items.
Current Item
Camera
Objects
Bones
Lights
Cameras
Item Properties
Auto Key
Create Key
Delete Key
Preview
Step

Camera Match plug-in interface, showing feature null names and their image coordinates.

Continued on page 28

YOUR NUMBER 1 SOURCE FOR DIGITAL VIDEO, ANIMATION, MULTIMEDIA AND WEB DEVELOPMENT

A Safe Harbor Exclusive!

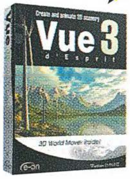
BUY
Lip Service™
& SHAVE AND A HAIRCUT™
IS FREE!!
Limited time offer!

Lip Service – A unique software package that enables animators to quickly and easily generate a multitude of facial expressions, mouth movements, and phonemes, all with realistic results! *Includes Shave And A Haircut* 286.00

Shave And A Haircut – Features include Shadowing, Motion Blur, Collision Detection and more.285.00

A Safe Harbor Exclusive!

Vue d' Esprit 3 – This powerful 3D application is optimized for creating, rendering and animating natural scenery. **Vue d' Esprit 3** was designed for the beginner, as well as the advanced user. It is fast, easy to use, and includes many new features, such as animation, elaborate timeline, and professional video options. Also, fully textured objects created with LightWave 3D can now be imported and exported easily 189.00



OXYGEN GVX1
3Dlabs

Oxygen GVX1- AGP
A 100% geometry and lighting acceleration card which integrates the GLINT R3, and the GLINT Gamma G1 processor on a single AGP board 699.00

Oxygen GVX1- PCI.................899.00

Oxygen GVX210 AGP – Delivers high-end geometry and rasterization graphics acceleration on a single AGP board 1349.00

Oxygen VX1 AGP – Achieve 2D and 3D superb graphics with the GLINT R3 processor 209.00

Oxygen VX1 PCI 199.00

DVRex-M1

Canopus DVRex-M1/Premiere - Combines high-quality DV video with multi-channel audio support; includes Premiere 5.1, Kick Start Training CD, breakout box, Rex Edit 2.5, and cables................................2399.00

Canopus Rex RT Option - Add real-time effects to your DVRex editing system!...........................999.00

EZOV.......................................249.00

Raptor Bay...................................99.00

Xplode.......................................399.00

DVRaptor - A complete DV camcorder editor which includes Adobe Premiere 5.1, Kick Start Training CD and all required cables...499.99

Mega Production Bundle - All-in-one super package includes DVRex-M1 breakout box, Rex RT option, Rex Edit 2.7, Adobe Premiere 5.1, Rex Navigator software, and all cables...2999.99

LIGHTWAVE PLUG-IN/ LIGHTWAVE RELATED SOFTWARE

3D Super Bundle (Art Recon)	529.00
AuraFX Vol. 4	99.99
Alpha Magic Wizard	199.00
Amazing Sci-Fi Textures	103.99
Animation Master 2000	269.99
Animation Master/Simbiont Bundle	319.00
BlowUp & Shatter For LightWave	89.99
Boris FX 5.0	399.00
Commotion 3.0	1995.00
Complete Effects Library	369.00
Creature Creator for LightWave	225.00
DarkTree Textures	289.00
Dinosaurs ROM CD PLUS!	60.00
FiberFactory3	269.00
Genesis V2 Pro	269.99
Graffiti	399.00
Hollywood FX	Call
KFramED	Call
Life Forms Studio 3.9	399.99
Light ROM Super Bundles	Call
MacroForm 2.0, Intel	289.00
ReelMotion 1.0	634.99
NatureFX	699.00
Particle Storm 2.0	489.00
Power CG Plus/Speed Razor	295.00
Project: messiah	549.00
Relativity	189.99
Seamless Textures 6 – Architectural Ornament	129.00
SurfacesROM	34.00
Tree Druid, Intel/Mac	169.00
Visual Effects CD Set	132.99
World Construction Set 5.0 Intel/Dec	659.00

THE JOY OF SIX!

LightWave [6]
The new version of the award-winning 3D animation system from NewTek! Powerful new tools include enhanced interactivity and scene editor, Skelegons, MetaNURBS in layout with point weighting, caustics, UV Mapping, surface editing in modeler and a whole lot more! (specifications subject to change without notice) LightWave 3D is the most-used animation system on the planet! What are **you** using? 2250.00

LightWave [6] Upgrades ....Call

Aura 2.0.......................................895.00

Aura 1.0 - 2.0 Upgrade..................295.00

Aura Powerhouse Bundle – Includes Aura and AuraFX combo set! 799.00

Calibar ....Call

Inspire 3D/LD_Tools Bundle 459.99

Visit our website
www.videotoasternt.com

Video Toaster NT
Never again compromise the quality of your video - lossless video editing is now available on your desktop! Introducing Video Toaster NT - the next generation, uncompressed digital video solution from NewTek. For only $2995, your complete post-production studio includes:
• Speed Razor (award-winning non-linear editing) • Aura (2D graphics, manipulation & compositing) • Lightwave (professional 3D animation)

Already own a Video Toaster, Flyer or Lightwave? Save $500 when you upgrade to the Video Toaster NT!

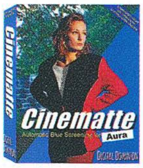
Complete reliable, quality turnkey Toaster workstations are also available!

Video Toaster NT SDI Option........995.00

Pre-order NOW!

Video Toaster NT 1.0-2.0 ....495.00

Video Toaster NT 1.0-2.0 + Breakout box 1995.00



Cinematte for Aura
New plug-in for Aura! The acclaimed blue screening tool that allows you to composite images together while maintaining proper anti-aliasing and transparency is now available!. Works with Aura 1.0 and 2.0.................239.00

dpsReality
The world's first Studio Digital Disk Recorder! Features compressed or uncompressed recording, alpha channel, balanced/unbalanced stereo audio I/O and DPS Virtual Tape File System for group rendering. The package contains the dpsReality player application, Digital Fusion DFX compositing software, and an audio/video breakout cable. 2199.00

Reality
Studio Digital Disk Recorder

DPS™

dpsVelocity – 2D
The combination of hardware and software is what makes this real-time NLE system the ultimate choice. The **dpsVelocity – 2D** delivers faster, sharper, and easier edits at an affordable price. This real-time editing solution includes the dpsReality card, dpsReality player, audio/video breakout box cable, dpsVelocity NLE software, and so much more 2699.00

dpsVelocity
FASTER, SHARPER, EASIER EDITS

dpsVelocity – 3D 4499.00

dpsVelocity NLE Software..........805.00

Visit our site
www.dvguide.sharbor.com/dps/
for more information.

Terms: POs accepted from schools and government agencies • No waiting for personal or business checks • Defective products replaced promptly. RMA number required (call 262-548-8159) for all merchandise returns. Returns not accepted after 15 days. Returned products must be in original packaging, postage prepaid. Opened software not returnable. Shipping charges not refundable. Returns subject to a 15% restocking fee.• Not responsible for typos. Prices subject to change.

Information: 262-548-8120
Fax 262-548-8130
Tech Support / RMAs:
262-548-8159, 12-4 pm Mon.-Fri.
Safe Harbor Computers
W226 N900 Eastmound Dr.,
Waukesha, WI 53186

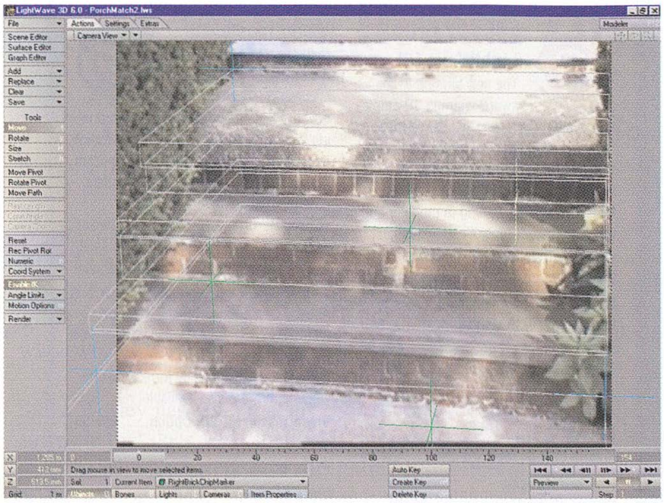
ORDER ON-LINE 24 HRS. A DAY

To Order Call 800-544-6599

MON.– FRI. 9-5:30 CST

SAFE HARBOR COMPUTERS

www.sharbor.com



Results of first pass with Camera Match, using five features.

take measurements at the set, and you no longer have access to the objects or fixed features that were photographed, don't bother trying to make a match. Your results depend on accurate measurements; no numbers, no results. Guesstimates? You're better off eyeballing a manual match.

Once you have your measurements, you need to define feature points by placing nulls at the measured locations. As a test, I used the same front porch object I described in my book's projects. I had already built the model, and had pretty accurate measurements of the visible artifacts I had used for tracking and stabilization. In Layout, I added nulls to each feature location. I opened the first image in Photoshop and used the Info panel to get the XY coordinates of each feature. Finally, I opened the Camera Match plug-in and typed in the data. The first pass produced a pretty good match, as shown above. Not bad for about 15 minutes' work; it also identified where my measurements were off, so I was able to make corrections for an even better match on other frames.

Camera Match is a tool you should not do without, if your work ever requires you to match LightWave elements with real-world images or footage.

Sticky Front Projection

You've probably used LightWave's Front Projection Image Map. It's very useful when you're trying to composite LightWave elements over a photograph or live-action footage. You simply model a few rough shadow-catching surfaces, apply the photo as a Front Projection, and render. However, moving the camera or the object requires you to perform an exact move match, a very expensive process.

StickyFP enables you to stick the image map to the surface, so when you move the object or the camera, the image map is still applied accurately. With this tool, you can add depth to make a 3D animation from a single image, hide problems in a less-than-perfect move match, or create photorealistic maps for an object to be used in another animation.

StickyFP is especially powerful when used with Camera Match. Apply Camera Match to nail down an accurate camera position, apply StickyFP to stick down the image map to the object's surface, and you have as accurately-mapped an object as possible. If your work requires matching live-action objects, StickyFP and Camera Match will be your new best friends. My only complaint is that I didn't have these plug-ins a year ago!

Whirley Points

Inertia works. In real life, nothing is perfectly rigid, and the faster you move something, the more it flops around. In character animation, this effect of inertia is called overlapping action. Anyone who has attempted to automate overlapping action (especially for the flesh or appendages of a complex character) by using LazyPoints or Motion Designer can attest to the trade-off between low quality results and difficult setups. Whirley Points is a nice compromise; it is relatively easy to set up, and the results are pretty good.

The most basic Whirley Points setup is to define the surfaces of your model according to their flexibility, then select the flexible surfaces by name. For example, if you break up a suit coat into rigid surfaces (the shoulders) and more flexible surfaces (the hemline), you can tell Whirley Points to affect only the surfaces listed in the Surface Names field. The Sharpness control affects the blend between affected and unaffected surfaces, so you can get smoothly graduated motion.

If you want the entire object to be affected, you can choose Apply to Entire Object. You can fine-tune the effect by adjusting the Effect Bias control; most commonly, you'll want the farther points moved most, and those near the center of the object to remain relatively rigid. The live preview shows the Effect Bias in color-coded wireframe, rigid parts in blue and the flexing parts in yellow.

I like this plug-in a lot. Even though its applications are limited (no collision detection, only one setting per object), for many wobbly objects or simple characters, this provides all the soft-body dynamics you need. Besides, it's so interactive that it's fun to play with!

Tracer

If you're animating shoot-'em-ups, you need this plug-in. Setting up and animating gunfire, whether for photorealistic composited effects or sci-fi energy weapons, is tedious and far from creative. Sure, each round fired is simple in and of itself, but multiply that by the rate of fire of automatic weapons and you've got a huge job on your hands. The Tracer pixel filter plug-in boils it all down to clearly named (and mostly animatable) parameters such as fire rate, speed, maximum distance, and accuracy, plus a set of seven parameters that control the visual appearance of the 'volumetric glowing projectiles' rendered by this plug-in. If you've seen any of Foundation Imaging's firefight sequences for Starship Troopers, this is the tool that made those shots cost-effective.

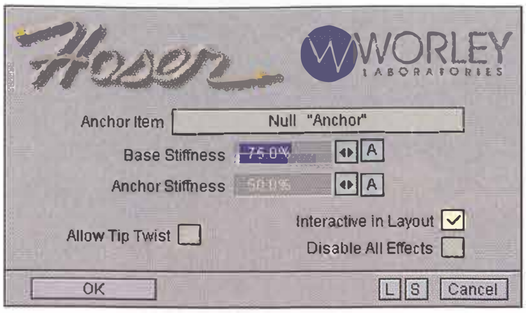
HeatWave

This is an image filter plug-in that recreates the turbulent distortion effects of hot air. You can apply it to an entire scene (think desert noontime), or limit its effects to specific surfaces. One demo scene shows a simple polygon, shaped and positioned as if a steam plume over a cooking pot on a stove. The HeatWave effect, limited to this polygon, leaves the rest of the scene unaffected but creates a fairly realistic shimmer above the pot. You should plan to experiment with the settings for this plug-in (as for all Worley plug-ins) to get the effect you want. Because this plug-in can mimic anything from a jet engine exhaust to a single match's heat plume, you have a lot of room for experimenting. This is especially important if you are trying to match or fake real-world footage; your audience may not be able to tell you, "The Vertical Waver was too high," but they will be able to say, "It just doesn't look right."

Hoser

This deformation plug-in can do some amazingly complex things for your animations, but it has one of the simplest interfaces I've ever seen. You simply choose your hose object, choose an anchor item (usually a null) to control the opposite end of the hose, and set the stiffness parameters. Hoser does the rest.

This plug-in has applications for nearly every LightWave user. If you do technical animations, Hoser enables you to string rigging, run hoses, connect springs, and otherwise build automatic setups that mimic real-world devices. If you need to simulate complex organic processes, you can use Hoser to set up tendons, ligaments, arteries and other tissues that will stretch and bend with the model's motion.



Anchor Item: Null "Anchor"
Base Stiffness: 75.0%
Anchor Stiffness: 50.0%
Allow Tip Twist ☐
Interactive in Layout ☑
Disable All Effects ☐

Hoser plug-in interface.

If you are a character animator, Hoser enables you to easily set up and animate the minimalist style known as 'rubber-hose animation', wherein the character's hands and feet are connected to its body with flexible hoses rather than rigidly jointed arms and legs. Pose the body, pose the hands and feet, and Hoser takes care of the rest. By chaining two or more hoses together, you can directly animate the middle joint. This enables you to define a knee or elbow to create a stronger pose, while preserving your options for a wiggly, apparently boneless limb.

Mosaic

This image filter plug-in renders a blocky grid or radial mosaic pattern over an area you define, simulating the effect of video censorship or scrambled digital transmissions. As with most Worley plug-ins, nearly all the parameters are animatable. The ability to confine the effect to the area of a null or other object makes this plug-in very easy to set up and use.

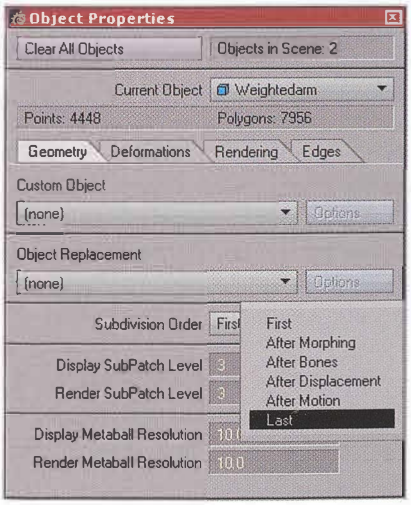
In Summary...

Depending on the nature of your LightWave work, you may use only a few of these plug-ins regularly. However, I'd bet that sooner or later, you'll use them all. The Taft Collection is a time-saving investment that will pay for itself very quickly. Highly recommended.

Continued from Page 7

because you can check the low-res preview for any glitches or mistakes that you may have missed while putting the scene together. If it takes your computer one full day to render a 320x240 preview animation with low anti-aliasing, then you can be pretty certain that it will take at least 100 hours to render a 640x480 full-size animation with medium anti-aliasing, and this will be a much more accurate estimate than any other method can produce.

Q I created a MetaNURBed endomorph object and used the MorphMixer plug-in to keyframe some simple morphs between targets in the endomorph, but when I render out the animation or even when I preview the morphing object shape, there seem to be many errors in the shape of the object's mesh. What am I doing wrong?



Objects in Scene: 2
Current Object: Weightedarm
Points: 4448
Polygons: 7956
Custom Object: (none)
Object Replacement: (none)
Subdivision Order: First
Display SubPatch Level: 3
Render SubPatch Level: 3
Display Metaball Resolution: 10.0
Render Metaball Resolution: 10.0

***Figure 4:** The various options available from the Subdivision Order pull-down list.*

A In the Objects panel, under the Geometry tab, you must make sure that the Subdivision Order option is set to "Last". This tells LightWave when to apply SubPatch divisions to a MetaNURBed object, in order to create the final smooth shape of the object. The default option of "First" will subdivide the object before MorphMixer has a chance to alter the shape of your object. Subdividing too early can often create render errors in the object's mesh.

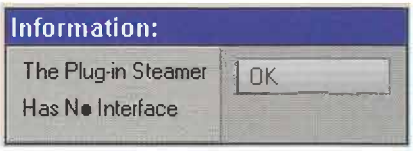
Q I have created a scene in L6 that uses the new features in HyperVoxels 3.0 to create some great-looking clouds with just a series of null objects. When I save the scene and then re-load it later, all of my HyperVoxel settings are lost. Why is this happening and how can I fix it?

A Unfortunately, you have discovered one of the few major bugs still present in the B update of L6. In fact, many people (like me) experience a complete crash of L6 when loading a scene that has HyperVoxels applied to null objects without any polygons. This is a known bug, and the odds are very good that by the time you read this, a new L6 update will be available. I feel pretty certain that this bug will be fixed in that update. Let's keep our fingers crossed, because it is one of the few remaining deadly bugs in L6.

Aside from a bug-fix, there are a few ways to work around this problem. The most obvious solution is to replace the null objects with some single-point polygons (particles) and apply your HyperVoxel settings to these particles. Note that in older versions of LightWave, you had to apply a HyperVoxel displacement plug-in to each object you wanted to have influenced by HyperVoxels. However, in L6 all you have to do is double-click on the appropriate object in the HyperVoxel interface, and the point data for that object will be automatically read in. Now when you save your scene file, the HyperVoxel settings will be saved correctly, and when you re-load the scene, everything will be just as you saved it.

Another simple solution would be to save your HyperVoxel settings and then re-apply them when you reload the scene. However, this solution would obviously not work if you were rendering your scene over a network. You could also create a new HyperVoxel preset by pressing "s" on the keyboard to save your current HyperVoxel settings to the Preset Library. Again, this does not solve the main problem of the HyperVoxel settings not being present when the scene is re-loaded.

Q I have discovered that I can add all of my old LightWave 5.6 plug-ins to L6, but when I try to access them, I get an error message saying that "The Plug-in X Has No Interface". Why is this happening?



The Plug-in Steamer Has No Interface

Figure 5: *Many of the old 'legacy' plug-ins will generate this error message when you try to open the plug-in's interface.*

A Many of the old LightWave plug-ins, also known as 'legacy' plug-ins, can be added to L6 so that your old 5.6 scenes and objects will load and render properly in L6. This gives L6 a certain degree of backward compatibility with older versions of LightWave, but many of these plug-ins will not allow you to make any changes to their settings. Initially, you may think that this is a major problem. However, you will find that many of the plug-ins that no longer work properly have been replaced with other alternatives in L6. For example, scenes that make use of Steamer will load and render correctly, but you cannot make changes to the Steamer settings. The Steamer plug-in has been replaced with built-in volumetric effects, so the only reason to have a plug-in like Steamer added to your L6 configuration files is to maintain backward compatibility with your old scene files.

Note that many, but not all, of the LightWave plug-ins created by third-party developers did not work correctly with L6 when it was first released. However, many plug-in developers have released updates that will make their products fully compatible with L6 and, in many cases, will take advantage of the new features in L6. If you have a third-party plug-in that you paid for, and it is not working correctly with L6, you should check the company's website to see if there is an update available. If you do not see an update available, you should contact the company to see what their plans are for supporting L6.

For those of you who enjoy reading this column and find a great benefit from the knowledge exchanged on the LightWave Mailing List, I encourage you to check out a new product my company has just released: The LightWave Mailing List Search Engine CD contains nearly 87,000 messages compiled together in a format that is easy to search for any topic that has been discussed in the past 3 years! It is available now for only $19.95 and is a valuable information resource for all LightWave users. For more info about David Warner, TutorVision 3-D CD, (his series of multimedia tutorial CD-ROMs for LightWave) and a FREE TutorVision Sampler CD, please visit http://www.tv3d.com or send some e-mail directly to Dave at dave@tv3d.com.

Continued from Page 32

mogulosity, can't you?). But is your desire strong enough to carry you through those 22-hour days? When you're on the fifth pot of coffee, there has to be something ingrained, subconscious, to keep you double-clicking.

As for me, I stopped animating a few years ago. It was a hobby for a long time, and when I got my first job animating professionally, the high lasted quite a while. But it didn't take. I found myself loathing something that had once given me tremendous satisfaction. When it was a hobby, I could just drop a project when it got boring.

However, that sort of work ethic is frowned upon when someone is paying you to finish said project. I still love special effects and animation, and am thankful that I was able to find a path traveling parallel to my original one. I spend my day with artists who love what they do (even if they don't always love every shot), and I get to catch a little bit of that energy every day. Sure, I may not be piloting spaceships anymore.

But I get a window seat on every flight.

Steve Pugh is the General Manager at Foundation Imaging. You can contact him through its website, www.foundation-i.com.

Continued from Page 12

notes its location with a hybrid path format, such as

`Z:images/image.iff`

or

`iMac:NewTek/Registration/images/image.jpg`

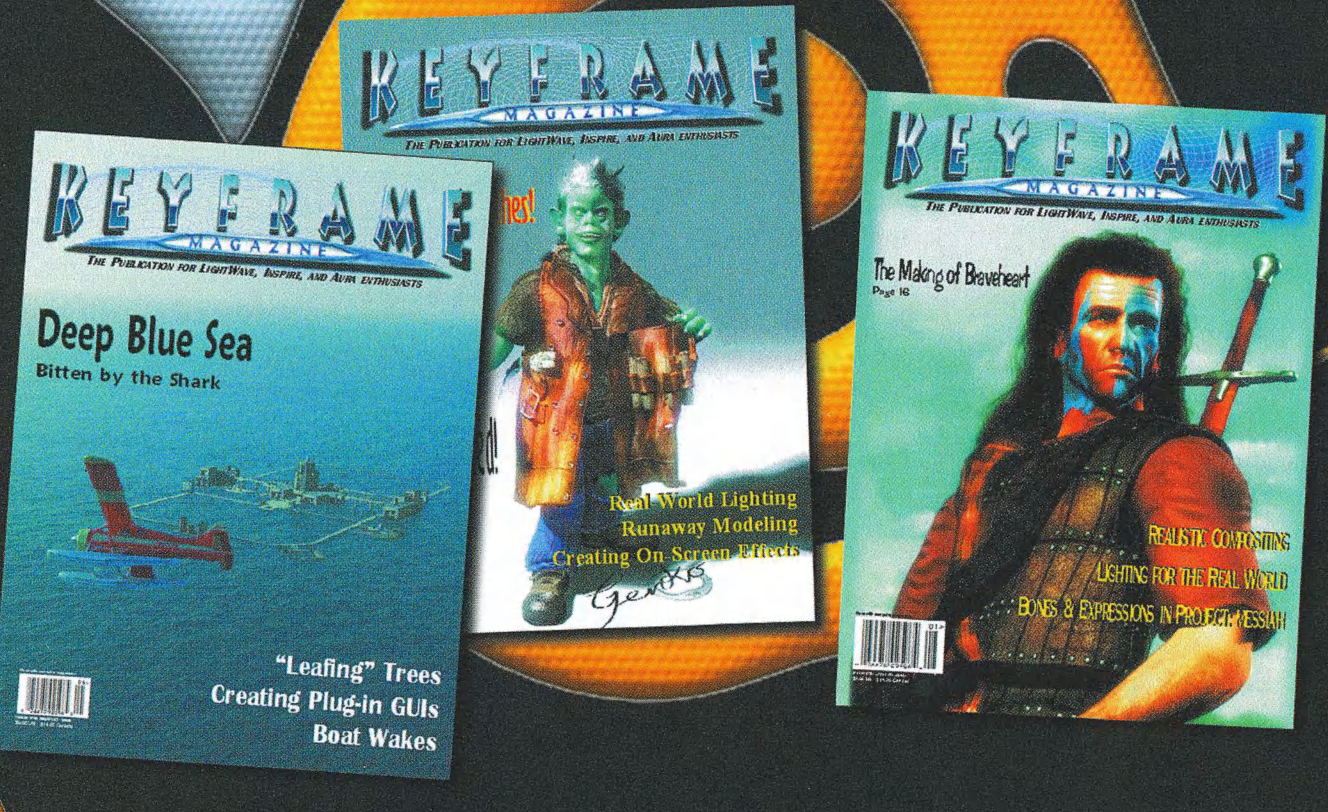
Both the Mac and PC version of LightWave read these paths correctly. This means that, except for the line ending problem, scenes could potentially be easily transported from Mac to PC and vice versa with significantly less problems than in 5.6. Additionally, this remains true when transferring scenes from removable media to your hard drive, or between folders and locations on your computer or network. This is a great improvement over LightWave 5.6's handling of paths, but it seems like an idea that has been only partially realized. Oddly, within the scene file itself, the path for saving rendered RGB and Alpha images is not in this hybrid format, but in the format of the platform on which the scene was saved.

All in all, Mac L[6] represents a huge step forward. My personal suggestion to NewTek is to concentrate less on new features and to focus more on honing what we have now.

Jason Goodman is the CEO of NYC-based Multimedia Software Design. You can email him at jason@msd-nyc.com or visit the website, www.msd-nyc.com.

Celebrate L6 With Us!

One FREE Issue to All New Subscribers and Renewals!


KEYFRAME MAGAZINE
The Publication for LightWave, Inspire, and Aura enthusiasts
Deep Blue Sea
Bitten by the Shark
"Leafing" Trees
Creating Plug-in GUIs
Boat Wakes
Real World Lighting
Runaway Modeling
Creating On Screen Effects
The Making of Braveheart
Page 16
Realistic Compositing
Lighting for the Real World
Bones & Expressions in Project: Messiah

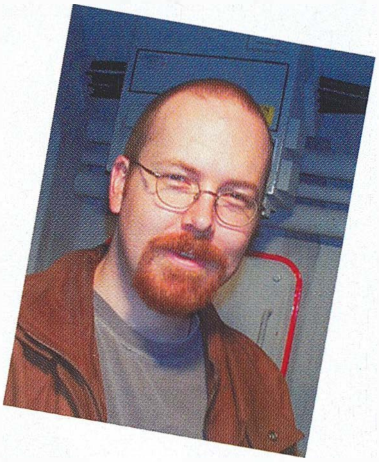
Back Issues Available

www.keyframemag.com

Free issue for new subscribers and renewals that are paid in advance with either a check, money order, Visa, Mastercard, or AmEx. Limited time offer.

Final Cut

by Steve Pugh



“Lights! Camera! Action!” Sounds romantic, doesn’t it? Glitz, glamour, all the things that put Hollywood on the map, can be summed up in that one phrase. And the glamour of the entertainment business draws thousands of people to its doors every year, looking for who knows what - fame, fortune, or just a chance to yell “That’s a wrap!”

There are as many reasons for getting into this business of 3D as there are artists. Some have wanted to fly spaceships since they were children. Others crossed over from traditional effects (but a lot of them probably wanted to fly spaceships anyway). I know there are some artists who animate because they just can’t help it. It’s easy for them, and of that I am jealous.

One thing I know is that the force driving these artists to animate for a living must be powerful. There are times when this thing we love to do becomes a chore, a job, work. (That ugly “w” word!) It may be awhile before it hits you, but it hits all of us sooner or later. This glamorous line of work we’re in eventually shows its true colors, and it’s a bit like the first time you find your parents putting Christmas presents under the tree. Maybe it’s the first time a client rips into you over a change you made to a camera move, maybe it’s the second or third time you find yourself transferring files in the office at 4 a.m. Either way, it’s no fun.

If you’re lucky, you can be thankful that you’ve got a talisman against the chore effect. If you truly love what you’re doing, you reach in and draw on that, and it gets you through. (Granted, a punching bag or a comfy place to crash can speed the recovery process greatly). You will only get into trouble if you don’t have that passion, that “core joy” in what you do. Ask yourself: “Why did you get into this?”

If all you want to do is direct, then bite the bullet and sally forth. The hours are longer and there will be plenty of people screaming at you down the road. Did you get into special effects to meet actresses? Bad news; there’s a reason it’s called POST-production. The closest you’re likely to come to an actress is a scanned element, and it’s just not the same.

Want to be a player, maybe? A mover and a shaker? Well, you sure picked the wrong end of the business to start in. We wear jeans and T-shirts at this end, thank you very much. Take your blazer and power tie over to the other side of the hill and get into a production office with the rest of the moguls. One thing’s for certain, the moving and shaking aren’t going to wander into some dimly lit, toy-encrusted animators’ den looking for you. We play with TOYS, for chrissake. (Allow me to self-edit briefly and suggest that some degree of mogulship may result if you’re really clever and make a short film about an airplane landing on a freeway.)

Granted, all of these things are possible side effects. Work hard enough and you’ll get the directing gig, maybe meet a nice actor who likes your smile. Fly enough spaceships and get your name around, maybe you’ll find yourself owning an effects facility and teetering on the edge of mogulness (you can tell I don’t know what to call

Continued on Page 30

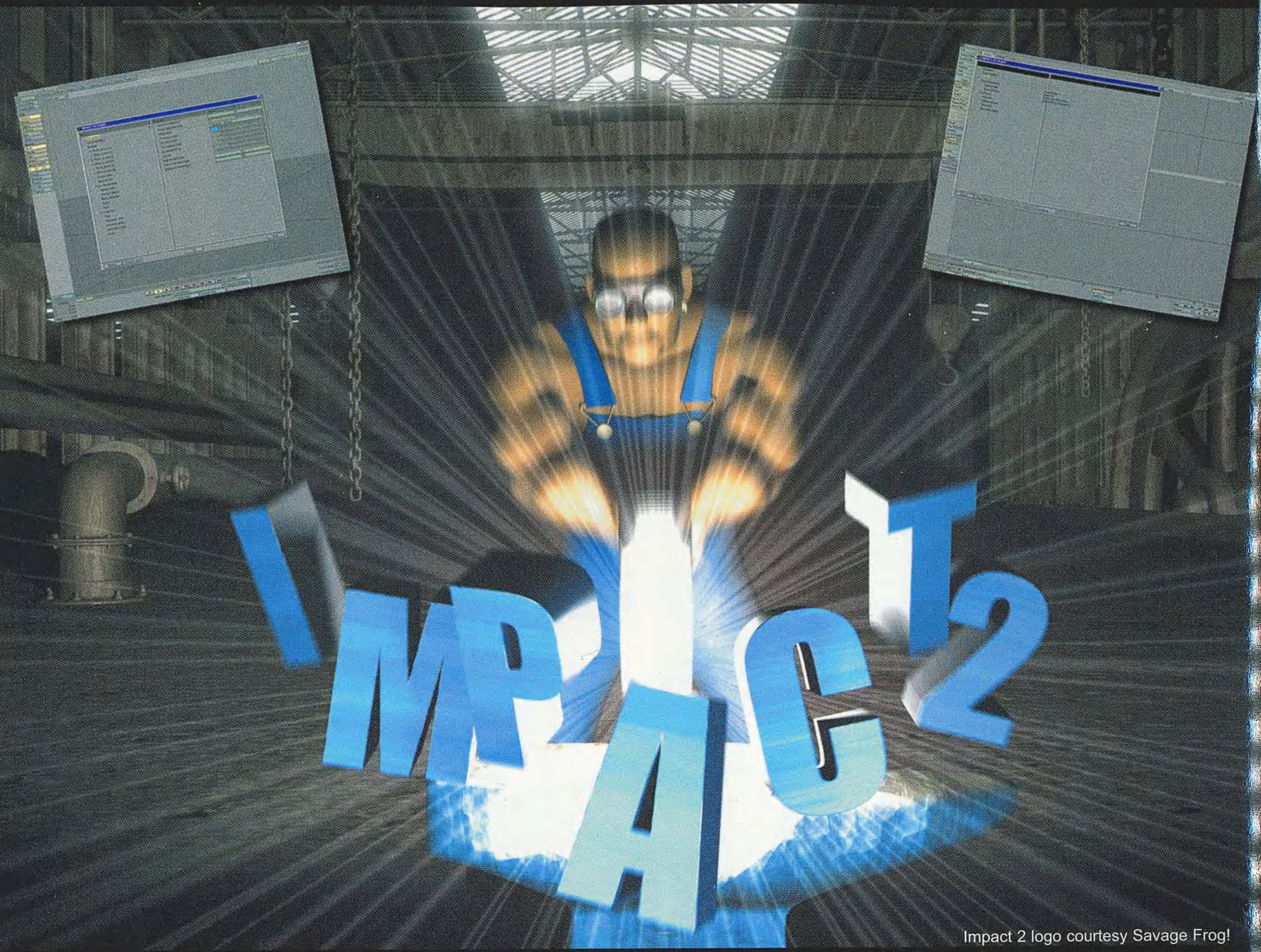
COMPRESSION
Sucks
The New
Video Toaster®

- Uncompressed D1
- SpeedRazor™ SE Non-linear Editor
- Aura™ Video Paint
- LightWave® VT

www.videotoaster.com
Call **(800) Toaster**
Outside the U.S. call (210) 370-8000

NewTek

Rigid Body Dynamics for LightWave 3D 5.6 and 6.0!



Impact 2 logo courtesy Savage Frog!

Breaking the barriers of rigid-body dynamics!

Completely redesigned and integrated in LightWave 5.6 and 6.0.

-Advanced collision detection and response-
-Several constraints such as spring, rods, pin joints, and more...
-Varying force generators-

Complete feature list and pricing at www.dynamic-realities.com

Dynamic Realities

Complete feature list and pricing at

www.dynamic-realities.com